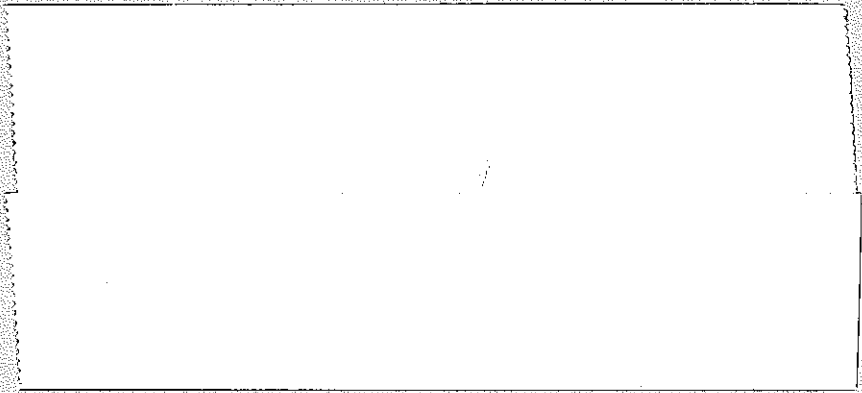


区分変更	
変更理由	＝
決着年月日	平成13年7月31日

テクニカルリソース管理システム システム仕様書



1988年7月



動力炉・核燃料開発事業団

大洗工学センター



社 内 一 般

PNC TN 9520 88-022 (2)

1 9 8 8 年 7 月

テクニカルリソース管理システム ” B A H A M A ” システム仕様書

BAHAMA W/G *

要 旨

本システム仕様書は、R & Dに於ける実験的研究や設計活動での計算機を利用した試行錯誤的な作業過程で多発するプログラムやデータ等、多種多様な計算機リソースの作成、更新及び削除等に関する情報を記録・管理することによる作業効率の向上を目的として開発されたシステム BAHAMA (BAsed HAndling data and program MAnagement system) の内部仕様に関して述べたものである。

本システム仕様書は、システムの動作環境（システム環境、使用ファイル、使用ライブラリ等）、ファイル構造及びプログラム仕様（共通ブロック、モジュール仕様）に関して説明したものであり、システム管理者が主として引用するものである。利用方法に関しては、テクニカルリソース管理システムBAHAMA利用手引書（PNC N 9 5 2 0 8 8 - 0 2 2 (1) ）を参照されたい。

* 動力炉・核燃料開発事業団 機器構造開発部 材料開発室

目 次

1. 目的、機能概要	1
2. 動作環境	3
3. 使用ライブラリ	3
4. ファイル仕様	4
5. 共通ブロック仕様	1 4
6. モジュール仕様	2 4
7. エラーメッセージ	6 7
8. プログラムリスト	6 8

図 表 目 次

< 図 >

図. 1 システム動作環境図	2
----------------	-------	---

< 表 >

表. 1 データベースの登録可能データ一覧表	1
表. 2 各サブシステム機能一覧表	2
表. 3 使用機番及びファイル一覧表	3
表. 4 使用ライブラリ一覧表	3

1. 目的、機能概要

当システムは、ラボラトリオートメーション分野での実験的研究や設計活動における試行錯誤的なデータ解析作業を支援する会話型のリソース情報管理システムであり、多種多様な実験データ及び解析プログラム並びに文献及び報告書等の所在情報を、データベースに登録することによって、作業中或いは作業後の重要リソースを集中管理し、実研部門、研究部門及び設計部門等の研究作業の能率向上を目的としている。

システムは、一般利用者用とシステム管理者用のサブシステムに分割されており、どちらのサブシステムも一括画面入力によるTSS会話型処理機能を有する。また登録者登録日、更新者及び更新日をシステムが自動登録するため、データ所有者が常に明らかである。データの更新及び削除時は、誤って他人のデータを処理しないようにレコードの登録ID或いは更新IDと操作中の利用者IDが一致しない場合は、処理パスワードの入力を促す等のデータ保護機能が動作し、かつ更新前及び削除前のレコードを自動的に保管するため誤って削除及び更新処理を実行してしまった場合でもレコードの回復が可能である。

データベースの登録可能データ及びサブシステム機能を表. 1及び表. 2に示す。

No	登録可能データ	文字数	備考
1	ファイル名	44	各種リソース名
2	ファイル種別	20	リソースの種別
3	データ登録者ID	5	データを登録した利用者ID (システムが自動登録する)
4	データ登録日	8	データを登録した日付 (システムが自動登録する)
5	データ更新者ID	5	データを更新した利用者ID (システムが自動登録する)
6	データ更新日	8	データを更新した日付 (システムが自動登録する)
7	検索キー	20×5	データ利用者の任意文字列
8	備考	50	保管場所、更新理由等の覚書

表. 1 データベースの登録可能データ一覧表

一般利用者サブシステム機能	管理者サブシステム機能
① データの登録	① 利用者の登録
② データの登録	② 利用者の更新
③ データの登録	③ 利用者の検索
④ データの登録	④ 利用者の削除
⑤ 検索結果のリスト出力	⑤ データ一覧リスト出力
	⑥ ログデータ出力
	⑦ 利用者一覧リスト出力
	⑧ ログファイルの初期化

表. 2 各サブシステム機能一覧表

システム環境図 ホストコンピュータ
 FACOM M780-10S OSN/F4 MSP

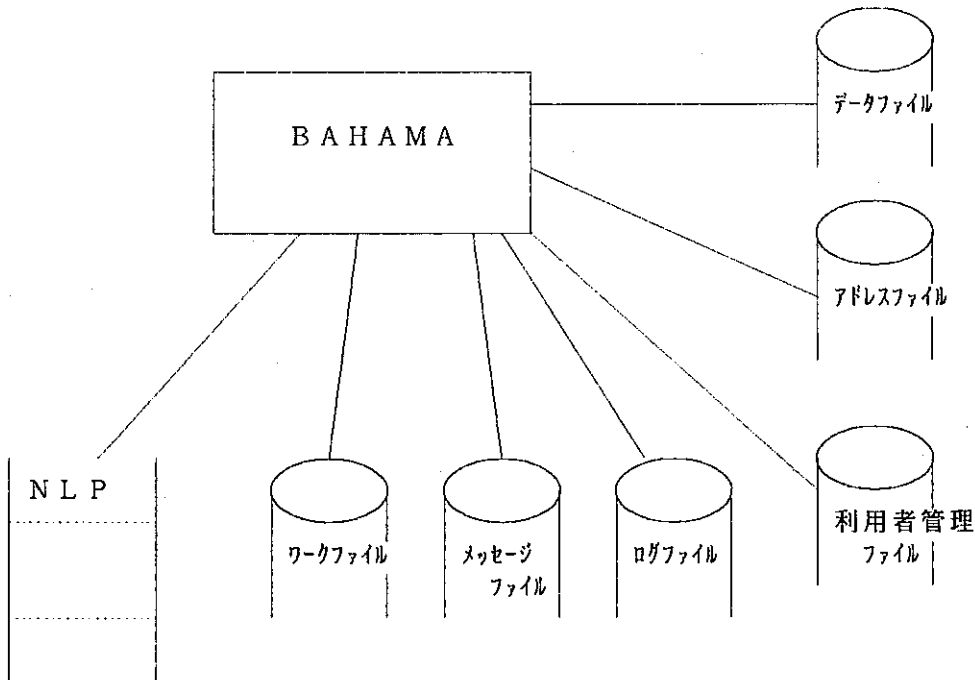


図. 1 システム動作環境図

2. 動作環境

論理機番	I/O	媒体	備考 (ファイル名、データセット名 等)
FT05F001	I	端末	端末入力機番
FT06F001	O	端末	端末出力機番
FT10F001	I/O	DASD	ユーザID. BAHAMA. DB
FT11F001	I/O	DASD	ユーザID. BAHAMA. PW
FT20F001	I/O	DASD	ユーザID. BAHAMA. ADDR
FT30F001	O	DASD	ユーザID. BAHAMA. LOG
FT40F001	I	DASD	ユーザID. BAHAMA. MSG
FT50F001	I/O	DASD	利用者ID. 当日日付. WORK
FT60F001	O	NLP	SYSOUT=G
FT70F001	I/O	DASD	ダミー
FT90F001	I/O	DASD	ユーザID. BAHAMA. PW
SYSMENU	I	DASD	IPF画面定義体 ユーザID. IPF. DATA

表. 3 使用機番及びファイル一覧表

3. 使用ライブラリ

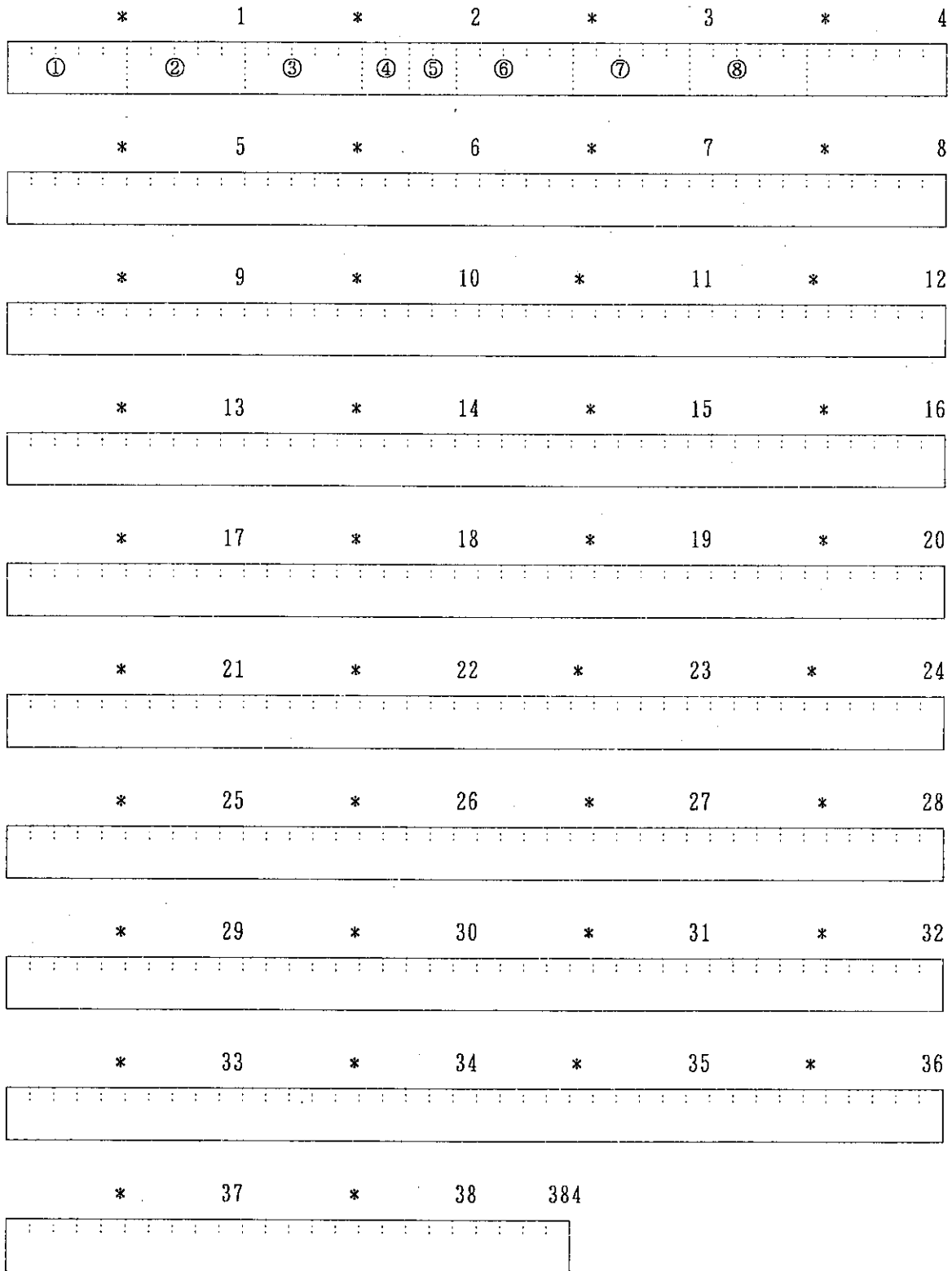
データセット名	内 容
SYS1. FORTLIB	フォートランライブラリ
SYS2. LINKLIB	IPFライブラリ
SYS9. PNC. LOAD	計算センタライブラリ
Q402#. AEG. LOAD	材料開発室ライブラリ

表. 4 使用ライブラリ一覧表

4. ファイル仕様

★ファイル名 : DB

★レコード名 : DB管理レコード

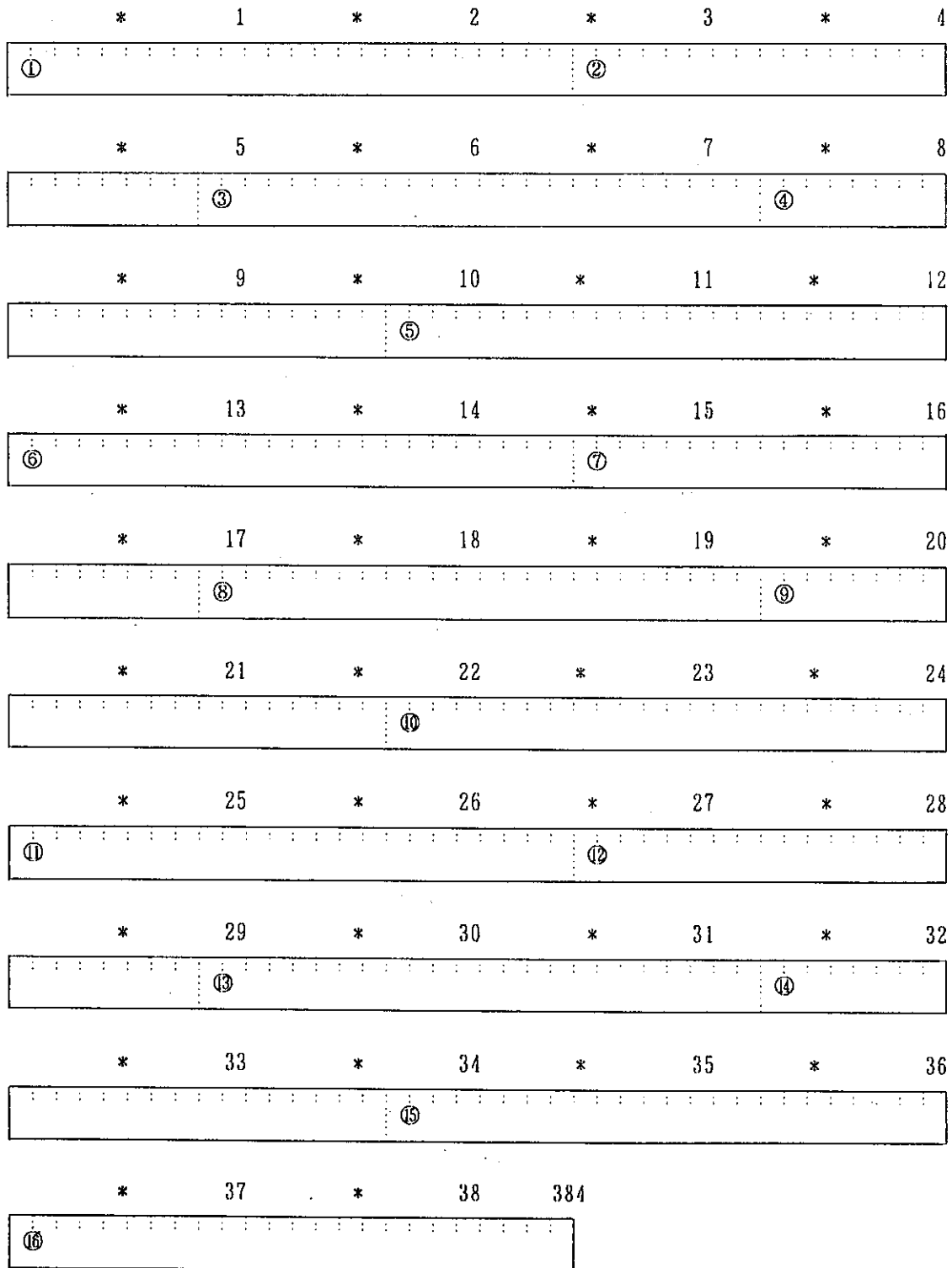


データの説明

- ① LRECDB (15) : DBレコード長 (384)
- ② LRMAX (15) : DB内レコード登録許容数 (32000)
- ③ LRNOW (15) : DB内レコード現登録数
- ④ ITMAX (12) : 項目登録許容数 (16)
- ⑤ ITNOW (12) : 項目現登録数
- ⑥ ITEMPS (13) : 空レコード管理レコード開始アドレス
- ⑦ ITEMPE (13) : 空レコード管理レコード終了アドレス
- ⑧ IDATAS (13) : データレコード開始アドレス

★ファイル名 : DB

★レコード名 : 項目書式レコード

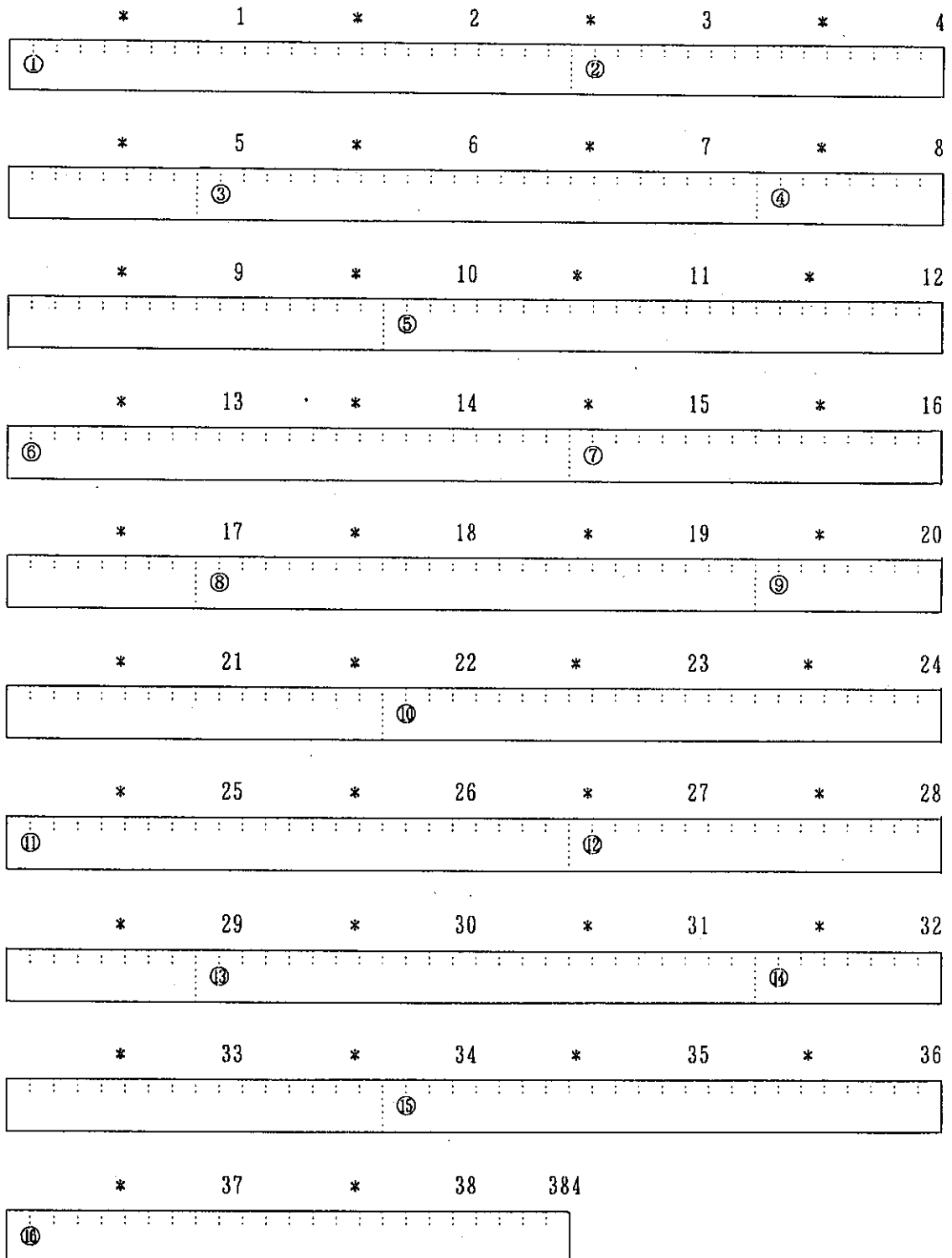


データの説明

①～⑯ ITFORM (A 2 4) : 項目書式 1～16

★ファイル名 : DB

★レコード名 : 項目書式レコード

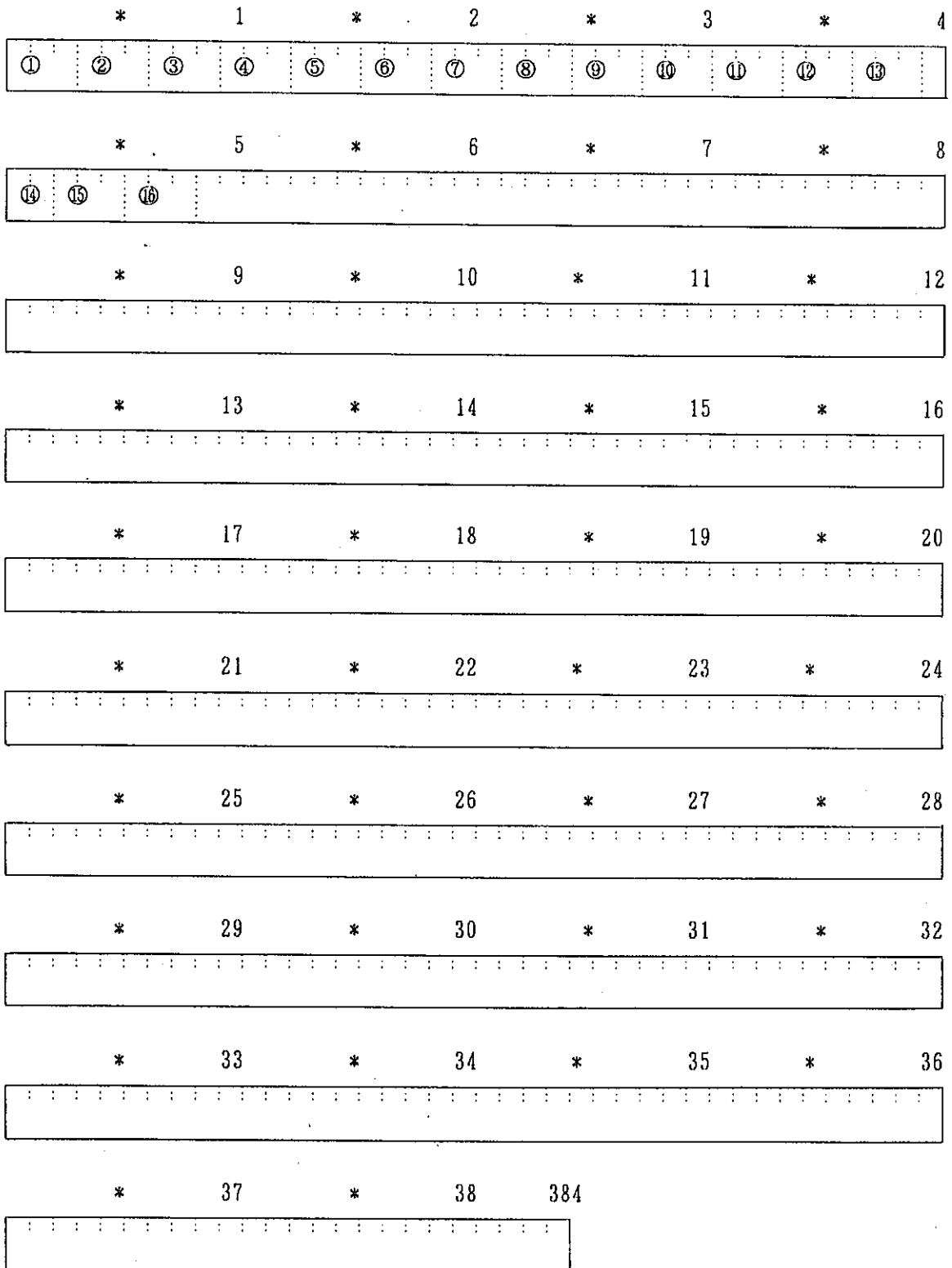


データの説明

①～⑯ ITFORM (A24) : 項目書式1～16

★ファイル名 : DB

★レコード名 : 項目文字長レコード

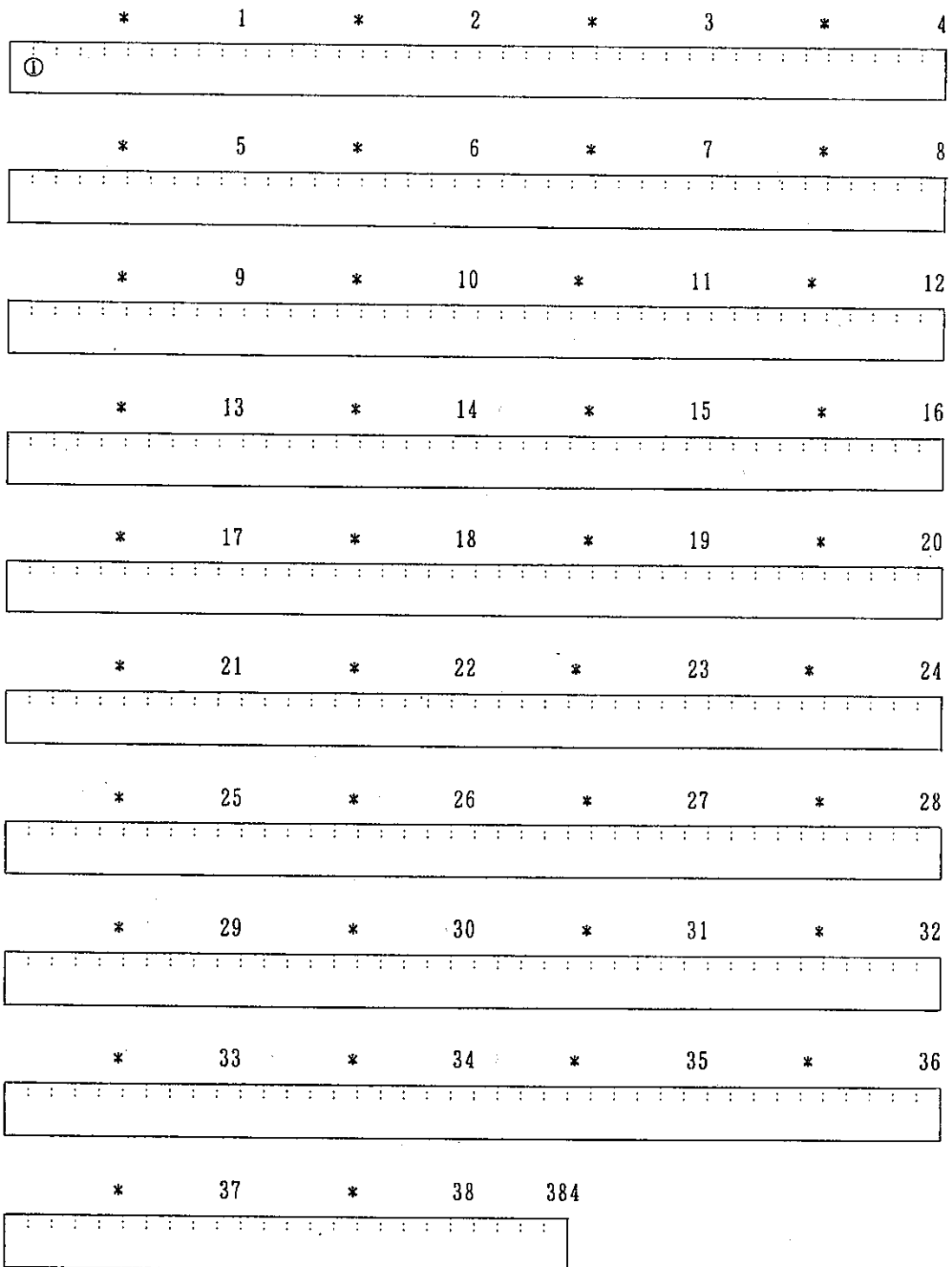


データの説明

①~⑯ I T C H A R (I 3) : 項目文字長 1 ~ 1 6

★ファイル名 : DB

★レコード名 : 空エリア管理レコード (1~11)



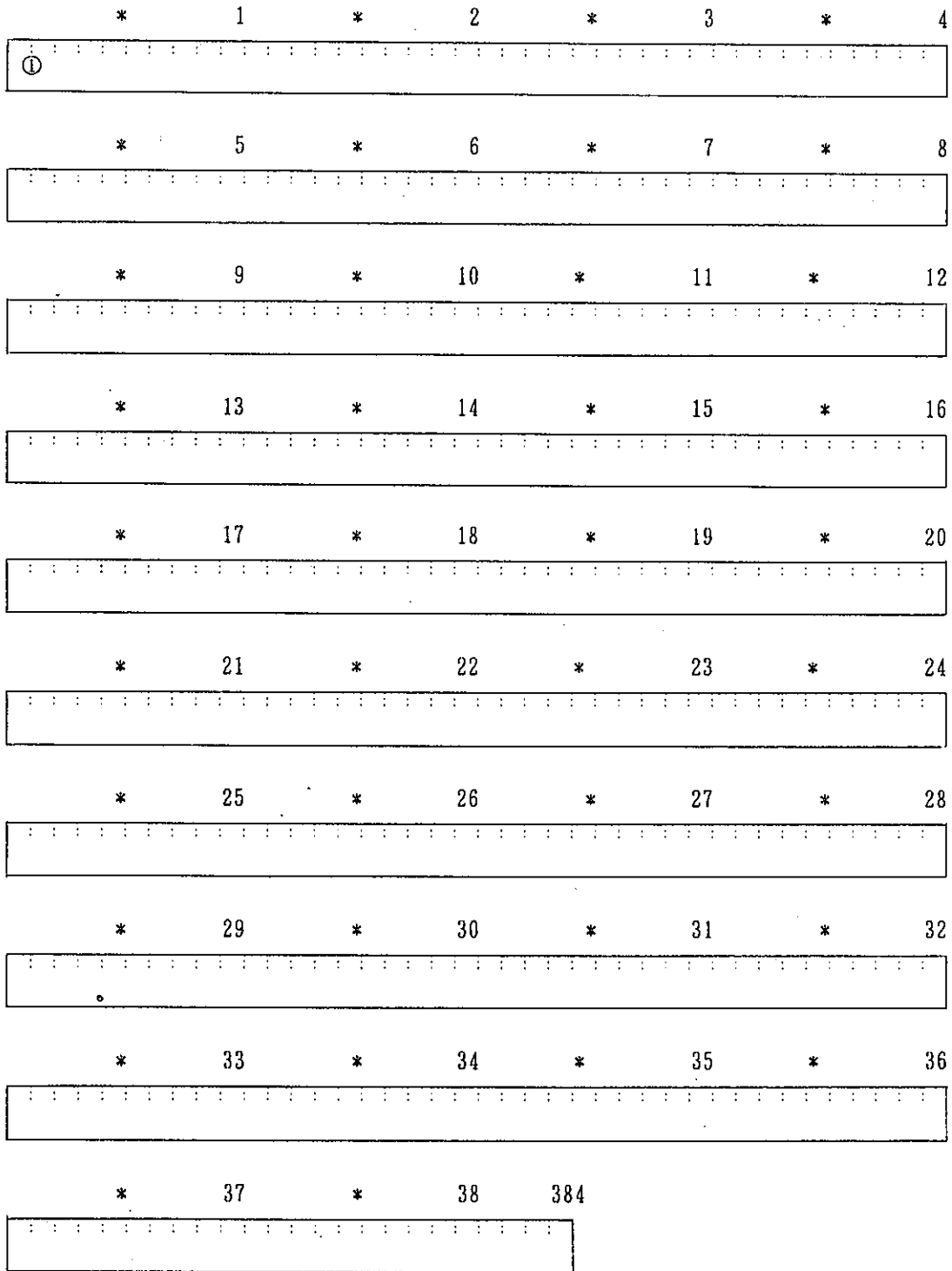
データの説明

① MAP : 空レコードの空状態を管理する

ビットON : 空き ビットOFF : レコードが存在する

★ファイル名 : DB

★レコード名 : データレコード

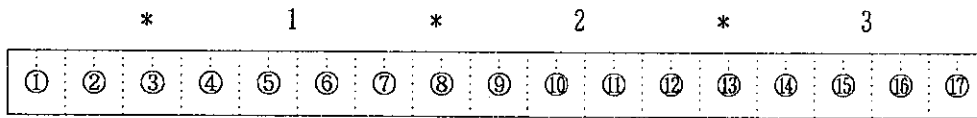


データの説明

① COND : ITNAME、ITFORM、ITCHARにより構成する
利用者データ

★ファイル名 : ADDR

★レコード名 : アドレスレコード

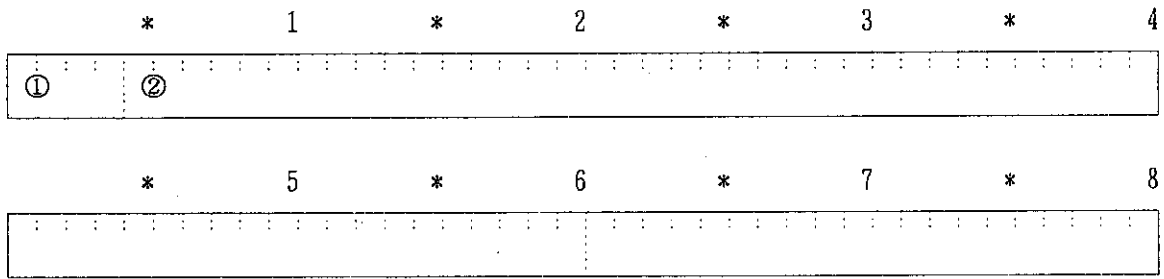


データの説明

- ① NDBAD : DB内アドレス
 - ②～⑰ NITAD : 各項目数字換算済データ
- 全て2バイトのレジスタ表現で構成される。

★ファイル名 : MSG

★レコード名 : メッセージレコード

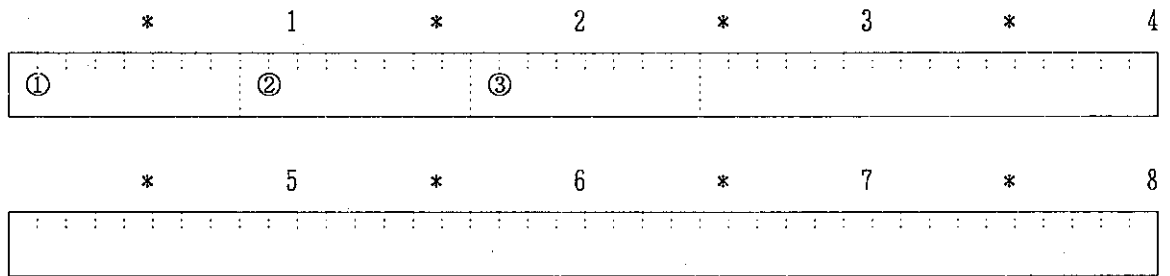


データの説明

- ① NERC D (I 4) : エラーコード
- ② MSG (A 6 0) : エラーメッセージ

★ファイル名 : PW

★レコード名 : 利用者管理レコード



データの説明

- ① UID (A 8) : 利用者ID
- ② UPW (A 8) : 一般利用者パスワード
- ③ OPW (A 8) : システム管理者パスワード

6. モジュール仕様

モジュール名称 : MAIN (BAHAMA)						
機能	重要ファイル管理システムのメインルーチンであり、パスワードチェック、機能選択及び処理対象ルーチンの呼び出しを行う。					
	変数名	型	大きさ	配列	I/O	内容
引数	メインなので無し					
特記事項	必要ファイル (DB, ADDR, LOG, MSG, WORK) が事前に割り当てられている事。					

モジュール名称 : ADDDB						
機能	コモン変数CONDITの内容に従ってDB及びADDRにレコードを追加する。					
	変数名	型	大きさ	配列	I/O	内容
引数	I ER	I	4		O	リターンコード
						0 : 正常終了
						1 : DB登録許容数を越えた
						2 : レコード管理エリアが矛盾している
特記事項	特に無し					

モジュール名称 : BAHAPW						
機 能	与えられた利用者IDの一般利用者パスワード及び管理者パスワードを得る。					
	変数名	型	大きさ	配列	I/O	内 容
引 数	LOGID	A	8		I	利用者ID
	UPW	A	8		O	一般利用者パスワード
	OPW	A	8		O	管理者パスワード
特 記 事 項	特に無し					

モジュール名称 : B A I N I T						
機 能	必要コモン変数に定数を設定するブロックデータルーチン					
	変 数 名	型	大 小 さ	配 列	I/O	内 容
引 数	無し					
特 記 事 項	特に無し					

モジュール名称 : BONOFF						
機能	与えられた変数領域内のビットのオン/オフ状態をチェックする。					
	変数名	型	大きさ	配列	I/O	内容
引数	IAREA	I	4	*	I/O	ビットのオン/オフチェックエリア
	ISB	I	4		I	チェック開始ビット相対位置
	IEB	I	4		I	チェック終了ビット相対位置
	ION	I	4		I	ビットのオン/オフチェック指示 0:OFFチェック 1:ONチェック
	IB	I	4		O	チェック指示に合致したビット相対位置
	IMODE	I	4		I	処理モード 0:ダミー 1:ビットリバース
	IER	I	4		O	リターンコード 0:正常終了 1:該当ビットが無い
特記事項	特に無し					

モジュール名称 : CHANUM						
機 能	英字で与えられた変数領域の内容を数字に変換する。					
引 数	変 数 名	型	大 き さ	配 列	I/O	内 容
	A	A	*		I	数字に変換する英字領域
	NUM	I	2		O	変換した数字設定領域
特 記 事 項	変換は、各1文字のビット構成によって行われる。					
	<p>例 Aを変換する場合</p> $A = A_{1(16)} = 01010001_{(2)} \text{ であるから}$ $A = 2^0 + 2^5 + 2^7 = 1 + 16 + 64 = 81_{(10)} \text{ となる。}$					

モジュール名称 : CHGDB						
機能	DBレコード及びADDRレコードの内容を置換し、置換履歴をログファイルに出力する。					
引数	変数名	型	大きさ	配列	I/O	内 容
	I E R	I	4		O	リターンコード
						0 : 正常終了
						1 : 置換データが選択されていない
特記事項	特に無し					

モジュール名称 : DBACCCE						
機能	① DBファイルをオープンし、データ管理レコードを入力する。 ② ADDRファイルを入力する。 ③ DBファイルをクローズする。					
引数	変数名	型	大きさ	配列	I/O	内容
	IOP	I	4		I	処理モード
						1 : DBをオープンし管理レコードを入力
						2 : DBをクローズする
	IER	I	4		O	リターンコード
特記事項	特に無し					

モジュール名称 : DBMENT						
機能	管理者用DBメンテナンス画面を表示し、指示された処理を実行する。 処理は、 ① DB内レコードのリスト出力 ② LOGファイルのリスト出力 ③ 利用者管理ファイルのリスト出力 ④ LOGファイルの初期化 である。					
	変数名	型	大きさ	配列	I/O	内容
引数	無し					
特記事項	特に無し					

モジュール名称 : DELDB						
機能	DB内のレコードを削除し、削除履歴をログファイルに出力する。					
引数	変数名	型	大きさ	配列	I/O	内容
	I E R	1	4		0	リターンコード
						0 : 正常終了
						1 : 処理対象レコードが選択されていない
特記事項	処理は、1件/1処理であり最初に選択されているレコードのみ対象とする。					

モジュール名称 : ERROR						
機能	指定されたメッセージ番号に相当するメッセージレコードを端末に出力する。					
	変数名	型	大きさ	配列	I/O	内容
引数	IERROR	I	4		I	メッセージ番号
特記事項	メッセージ番号は、メッセージ識別子（"BA"）を含んではならない。					

モジュール名称 : GETDB						
機能	データベースファイルを指定された方向に任意の位置から検索し、レコードを入力する。					
	変数名	型	大きさ	配列	I/O	内容
引	I START	I	4		I	検索開始アドレス (相対アドレス)
	I DIM	I	4		I	検索方向 + N : 終端方向に N 間隔で検索 - N : 先端方向に N 間隔で検索
数	I END	I	4		O	
	I CO	I	4		O	
	I RAD	I	4	20	O	
	I ER	I	4		O	
特記事項	検索結果はバッファ/INOUT/のBUFF¥W (1~18) に入力される。					

モジュール名称 : GETMSG						
機 能	メッセージファイルからエラーメッセージレコードを入力する。					
引 数	変数名	型	大きさ	配列	1/0	内 容
	無し					
特 記 事 項	特に無し					

モジュール名称 : H E D E R						
機 能	検索結果表示画面のリスト出力機能 (P F K 1 2) のリストヘッダ部を出力する。					
	変 数 名	型	大きさ	配列	I/O	内 容
引 数	I N	I	4		I	利用者識別フラグ
						1 : 一般利用者
						2 : システム管理者
特 記 事 項	特に無し					

モジュール名称 : LABEL						
機能	検索結果表示画面のリスト出力機能 (PFK12) のレコードラベル部を出力する。					
引数	変数名	型	大きさ	配列	I/O	内 容
	IN	I	4		I	利用者識別フラグ
						1 : 一般利用者
						2 : システム管理者
特記事項	特に無し					

モジュール名称 : LISTUP						
機能	検索結果表示画面のリスト出力機能 (PFK12) を実行する。					
	変数名	型	大きさ	配列	I/O	内容
引数	IRC	I	4		O	リターンコード
						0 : 正常リターン
						1 : 指定された機能が一般利用者機能及びシステム管理者機能以外である
特記事項	特に無し					

モジュール名称 : LOGOUT						
機能	レコードの削除及び更新に関するログレコードをログファイルに出力する。					
	変数名	型	大きさ	配列	I/O	内容
引	IFL	I	4		I	処理フラグ
						1 : 削除 2 : 更新
数	AREA	A	*(*)		I	ログファイル出力レコード
	IER	I	4		O	リターンコード
						0 : 正常終了
特記事項	特に無し					

モジュール名称 : MAGPUT						
機能	データベースのレコード管理エリアを所定のアドレスに出力する					
	変数名	型	大きさ	配列	I/O	内容
引数	IAD	I	4		I	処理レコード相対アドレス
	IER	I	4		O	リターンコード 0 : 正常終了 1 : IADがレコード範囲を超えている
特記事項	特に無し					

モジュール名称 : MATCHC						
機 能	利用者指定の条件に対しデータベースレコードの合致レベルを設定する。 検索条件数と合致レベルが同数になれば検索条件を満足したことになる。					
引 数	変 数 名	型	大 小 さ	配 列	1/0	内 容
	無し					
特 記 事 項	特に無し					

モジュール名称 : MATCHS						
機能	<p>指定された検索条件を項目毎に数値に変換し、検索条件の有無を設定する。</p> <p>項目毎の検索条件は / ADDR ¥ R / の I P R O に設定され、</p> <p>検索条件無し : . F A L S E .</p> <p>検索条件有り : . T R U E .</p> <p>が設定される。</p>					
	変数名	型	大きさ	配列	I/O	内 容
引数	無し					
特記事項	<p>共通ブロック / I N O U T / の C O N D を処理対象とする。</p>					

モジュール名称 : MENU						
機 能	システム管理者機能選択画面を表示する。					
引 数	変数名	型	大きさ	配列	I/O	内 容
	無し					
特 記 事 項	特に無し					

モジュール名称 : NUMSET						
機 能	/ INOUT / の COND に設定されたレコードを項目毎に数値化する。					
引 数	変 数 名	型	大 小 さ	配 列	I/O	内 容
	無し					
特 記 事 項	特に無し					

モジュール名称 : P ¥ W ¥						
機能	システム利用パスワードを確認する。					
引数	変数名	型	大きさ	配列	I/O	内 容
	TPW	A	8		I	端末から入力されたパスワード
	IRC	I	4		O	リターンコード
						0 : 正常終了
						1 : パスワードの問い合わせに解答した
特記事項	特に無し					

モジュール名称 : PAGE						
機 能	検索結果画面のリスト出力機能のページ及び実行日付を出力する。					
	変数名	型	大きさ	配列	I/O	内 容
引 数	ITOT	I	4		I	リストの総出力ページ数
	IP	I	4		I	現在の出力ページ番号
	DT	A	8		I	実行日付
特 記 事 項	特に無し					

モジュール名称 : RECCHK						
機能	検索条件を満足したレコードの内容を確認する。 (メモリ内では各項目毎の数値化データにより合致レベルを設定しているため 検索条件を満足したレコードのみがデータベースから入力され内容確認される)					
	変数名	型	大きさ	配列	I/O	内容
引数	無し					
特記事項	/INOUT/ BUFFW(20)をレコード入力のワークとして使用している					

モジュール名称 : UADD						
機能	一般利用者を利用者管理ファイルに登録する。					
	変数名	型	大きさ	配列	I/O	内容
引数	WORK	A	50	3	I	TSSコマンドエリア
						1: COPYコマンド
						2: ALLOCコマンド (WORK)
						3: ALLOCコマンド (PW)
特記事項	特に無し					

モジュール名称 : UDELET						
機能	一般利用者を利用者管理ファイルから削除する。					
引数	変数名	型	大きさ	配列	I/O	内 容
	WORK	A	50	3	I	TSSコマンドエリア
						1: COPYコマンド
						2: ALLOCコマンド (WORK)
						3: ALLOCコマンド (PW)
特記事項	特に無し					

モジュール名称 : ULIST						
機能	一般利用者一覧をNLPに出力する。					
	変数名	型	大きさ	配列	I/O	内容
引数	WORK	A	50	3	I	TSSコマンドエリア
						1: COPYコマンド
						2: ALLOCコマンド (WORK)
						3: ALLOCコマンド (PW)
特記事項	特に無し					

モジュール名称 : USERCH						
機能	一般利用者の利用者IDをキーにして利用者管理ファイルから検索する。					
引数	変数名	型	大きさ	配列	I/O	内 容
	WORK	A	50	3	I	TSSコマンドエリア
						1: COPYコマンド
						2: ALLOCコマンド (WORK)
						3: ALLOCコマンド (PW)
特記事項	特に無し					

モジュール名称 : USMENT						
機能	一般利用者管理画面を表示する。					
引数	変数名	型	大きさ	配列	I/O	内容
	無し					
特記事項	特に無し					

モジュール名称 : UUPDAT						
機能	一般利用者の利用者管理ファイル情報を更新する。					
引数	変数名	型	大きさ	配列	I/O	内 容
	WORK	A	50	3	I	TSSコマンドエリア
						1: COPYコマンド
						2: ALLOCコマンド (WORK)
						3: ALLOCコマンド (PW)
特記事項	特に無し					

モジュール名称 : DBMENT						
機 能	DBF (データベースファイル) 管理画面を表示し、DBF 管理機能を実行する。					
引 数	変 数 名	型	大 き さ	配 列	I/O	内 容
	無し					
特 記 事 項	特に無し					

モジュール名称 : ADD						
機能	登録データ設定画面で入力されたデータを登録する。					
	変数名	型	大きさ	配列	I/O	内 容
引数	COND	A	384		I	登録データエリア
	LOGID	A	8		I	利用者ID
	PDATE	A	8		I	処理日付
特記事項	登録処理は、原則として一件/処理とする。					

モジュール名称 : BUFFCL						
機 能	コモン変数 INOUT内をクリアーする。					
引 数	変 数 名	型	大きさ	配列	I/O	内 容
	BUFFYW	A	384	20	O	作業用エリア (画面入出力等)
	BUFF	A	1312		O	入出力バッファ (画面入出力等)
	COND	A	384		O	検索条件指定エリア (検索レコード設定)
特 記 事 項	特に無し					

モジュール名称 : DELETE						
機 能	データ選択画面でセレクト”S”を指定されたレコードを削除する。					
引 数	変数名	型	大きさ	配列	I/O	内 容
	LOGID	A	8	20	I	利用者ID
	LRNOW	I	2		I	DBファイルのレコード現登録数
	ISEL	L	1	32,000	I	画面でのセレクトフラグ
特 記 事 項	利用者IDと削除するレコードの登録者IDが異っている場合は、登録者IDのパスワードを画面入力しなければならない。					
	削除処理は、原則として一件/処理とする。					

モジュール名称 : PROC						
機能	<p>コモン変数PROSS・TERM・INOOUTを受け取り、登録・検索・更新・削除などの条件を入力し、それらの処理対象ルーチンの呼び出しを行う。</p>					
	変数名	型	大きさ	配列	I/O	内容
引数	IMISS	I	2		I	処理指示フラグ
	ITYPE	I	4		I	端末タイプ
	BUFF	A	1312		IO	入出力バッファ（画面入出力等）
	BUFF¥W	A	384	20	I	作業用エリア（画面入出力等）
特記事項	特に無し					

モジュール名称 : SEARCH						
機能	画面に設定された検索条件により検索処理を行う。					
	変数名	型	大きさ	配列	I/O	内容
引数	IUSYOW	I	2		I	利用者選別フラグ
	IMISS	I	2		I	処理別フラグ
	BUFFYW	A	384	20	IO	作業用エリア(画面入出力等)
	BUFF	A	1312		I	入出力バッファ(画面入出力等)
	COND	A	384		I	検索条件指示エリア(検索レコード設定)
	NDOM	I	2		I	条件合致レコード数
	IS	I	2	16	I	各項目の開始バイト位置
	IE	I	2	16	I	各項目の終了バイト位置
特記事項	特に無し					

モジュール名称 : SELECT						
機能	登録・検索・更新・削除の処理フラグを立てる。					
引数	変数名	型	大きさ	配列	I/O	内 容
	IMISS	I	2		I	処理別フラグ
	ITYPE	I	4		I	端末タイプ
	BUFF¥W	A	384	20	IO	作業用エリア（画面入出力等）
	BUFF	A	1312		I	入出力バッファ（画面入出力等）
特記事項	特に無し					

モジュール名称 : SEL1						
機能	<p>検索結果のレコードをmax18件/1画面に出力する。</p> <p>更新又は削除処理の時には、画面上で検索結果レコードを選択して、それぞれの処理を行うことができる。</p>					
	変数名	型	大きさ	配列	I/O	内容
引数	IMISS	I	2		I	処理別フラグ
	NDOM	I	2		I	処理指示フラグ
	ISEL	L	1	32,000	O	画面でのセレクトフラグ
	ITYPE	I	4		I	端末タイプ
	BUFF¥W	A	384	20	IO	作業用エリア(画面入出力等)
	BUFF	A	1312		I	入出力バッファ(画面入出力等)
	ISTART	I	4		O	検索開始アドレス(相対アドレス)
	IDIM	I	4		O	検索方向
	IEND	I	4		I	検索終了アドレス(相対アドレス)
	ICO	I	4		I	検索数(max18)
	IRAD	I	4		I	各検索済レコードの相対アドレス
特記事項	特に無し					

モジュール名称 : SEL2						
機能	検索結果のレコードをmax18件/1画面に出力する。					
	変数名	型	大きさ	配列	I/O	内容
引数	IMISS	I	2		I	処理別フラグ
	NDOM	I	2		I	処理指示フラグ
	ISEL	L	1	32,000	O	画面でのセレクトフラグ
	ITYPE	I	4		I	端末タイプ
	BUFF¥W	A	384	20	IO	作業用エリア(画面入出力等)
	BUFF	A	1312		I	入出力バッファ(画面入出力等)
	ISTART	I	4		O	検索開始アドレス(相対アドレス)
	IDIM	I	4		O	検索方向
	IEND	I	4		I	検索終了アドレス(相対アドレス)
	ICO	I	4		I	検索数(max18)
	IRAD	I	4		I	各検索済レコードの相対アドレス
特記事項	特に無し					

モジュール名称 : SEL 3						
機 能	検索結果のレコードをmax 18件 / 1画面に出力する。					
引 数	変 数 名	型	大きさ	配列	I/O	内 容
	IMISS	I	2		I	処理別フラグ
	NDOM	I	2		I	処理指示フラグ
	ISEL	L	1	32,000	O	画面でのセレクトフラグ
	ITYPE	I	4		I	端末タイプ
	BUFF¥W	A	384	20	IO	作業用エリア (画面入出力等)
	BUFF	A	1312		I	入出力バッファ (画面入出力等)
	ISTART	I	4		O	検索開始アドレス (相対アドレス)
	IDIM	I	4		O	検索方向
	IEND	I	4		I	検索終了アドレス (相対アドレス)
	ICO	I	4		I	検索数 (max 18)
	IRAD	I	4		I	各検索済レコードの相対アドレス
特 記 事 項	特に無し					

モジュール名称 : SEL 4						
機 能	検索結果のレコードをmax 18件 / 1画面に出力する。					
引 数	変数名	型	大きさ	配列	I/O	内 容
	IMISS	I	2		I	処理別フラグ
	NDOM	I	2		I	処理指示フラグ
	ISEL	L	1	32,000	O	画面でのセレクトフラグ
	ITYPE	I	4		I	端末タイプ
	BUFF¥W	A	384	20	IO	作業用エリア (画面入出力等)
	BUFF	A	1312		I	入出力バッファ (画面入出力等)
	ISTART	I	4		O	検索開始アドレス (相対アドレス)
	IDIM	I	4		O	検索方向
	IEND	I	4		I	検索終了アドレス (相対アドレス)
	ICO	I	4		I	検索数 (max 18)
	IRAD	I	4		I	各検索済レコードの相対アドレス
特 記 事 項	特に無し					

モジュール名称 : UPDATE						
機能	データ選択画面でセレクト”S”を指定されたレコードを更新する。					
引数	変数名	型	大きさ	配列	I/O	内容
	LOGID	A	8	20	I	利用者ID
	LRNOW	I	2		I	DBファイルのレコード現登録数
	ISEL	L	1	32,000	I	画面でのセレクトフラグ
	IS	I	2	16	I	各項目の開始バイト位置
	IE	I	2	16	I	各項目の終了バイト位置
	JC	I	2	3	I	プログラム内での処理指示フラグ
	BUFF¥W	A	384	20	I	作業用エリア(画面入出力等)
	BUFF	A	1312		I	入出力バッファ(画面入出力等)
	COND	A	384		I	検索条件指示エリア(検索レコード設定)
ISO	I	4		I	更新されるレコードの相対アドレス	
特記事項	利用者IDと更新するレコードの登録者IDが異っている場合は、登録者IDのパスワードを画面入力しなければならない。					
	更新処理は、原則として一件/処理とする。					

メッセージ ID	メッセージ内容	意味	対処
BA1001	END OF BASE DATA HANDRING MANAGEMENT SYSTEM	BAHAMAシステムを終了した	特に無し
BA1002	PLEASE SELECT THE PROCESS FLAG IN MENTE. PICTURE	管理機能選択が成されていない	目的機能を選択する
BA1003	IPF OPEN ERROR IN SUBROUTINE MENU	会話処理のオープンエラーが発生した	システム管理者に連絡する
BA1004	IPF INPUT/OUTPUT ERROR IN SUBROUTINE MENU	会話処理でI/Oエラーが発生した	処理を継続する
BA1005	SYSTEM PASSWORD & YOURE PASSWORD UNMATCHED	システム起動用パスワードが間違っている	正しいパスワードを入力する
BA1006	YOU CAN NOT USE IN OWNER SYSTEM PASSWORD UNMATCHED	—	—
BA1010	UN SUPORTED THIS TERMINAL FOR TTY INTER-FACE	TTY端末で当システムを起動した	TTY以外の端末から起動する
BA1011	IPF INPUT OUTPUT ERROR IN SUBROUTINE USMENT	会話処理でI/Oエラーが発生した	処理を継続する
BA1012	PLEASE TYPE IN USER ID	利用者管理機能で利用者IDの指定がない	利用者IDを指定する
BA1013	PLEASE TYPE IN USER PASSWORD OR OWNER PASSWORD	利用者管理機能でパスワードの指定がない	パスワードを指定する
BA1014	DUPRICATE USER ID ALREADY IN PASSWORD FILE	登録済の利用者を再登録しようとした	更新処理で実行する
BA1015	ADDED TO PASSWORD FILE	新規に利用者を登録した	特に無し
BA1017	DONT SAME PASSWORD IN USER AND OWNER FIELD	利用者パスワードと管理者パスワードが同一	利用者/管理者のパスワードを別にする
BA1018	THIS USER ID IS NOTHING IN PASSWORD FILE	指定された利用者IDは登録されていない	特に無し
BA1019	IPF INPUT OUTPUT ERROR IN SUBROUTINE DBMENT	会話処理でI/Oエラーが発生した	処理を継続する
BA1020	CHANGED TO PASSWORD FILE NORMAL END	利用者情報を更新した	特に無し
BA1021	DELETED USER IN PASSWORD FILE NORMAL END	利用者情報を削除した	特に無し
BA1022	INITIARIZED TO LOG FILE NORMAL END	ログファイルを初期化した	特に無し
BA2001	IPF INPUT OUTPUT ERROR IN SUBROUTINE SELECT	会話処理でI/Oエラーが発生した	処理を継続する
BA2002	IPF INPUT OUTPUT ERROR IN SUBROUTINE PROCC	会話処理でI/Oエラーが発生した	処理を継続する
BA2101	THIS RECORD IS ALREADY CATALOGED(ABNORMAL END)	指定したファイル名がすでに存在する	更新処理で実行する
BA2102	RECORD OVERFLOW IN DATA BASE	DBの最大値32000を超えた	システム管理者に連絡する
BA2103	RECORD MANAGEMENT ERROR	DBの入出力でエラーが発生した	システム管理者に連絡する
BA2104	DATA ADDED TO DB FILE NORMAL END	レコードを登録した	特に無し
BA2105	PLEASE INPUT THE FILE NAME IN ADDITIONAL PICTURE	会話処理でI/Oエラーが発生した	処理を継続する
BA2201	PLEASE INPUT THE SEARCH KEY IN SEARCH PICTURE	会話処理でI/Oエラーが発生した	処理を継続する
BA2202	DATA NOTHING IN DATA BASE	指定したデータがDB内がない	特に無し
BA2203	IPF INPUT/OUTPUT ERROR IN SUBROUTINE SEL1	会話処理でI/Oエラーが発生した	処理を継続する
BA2204	IPF INPUT/OUTPUT ERROR IN SUBROUTINE SEL2	会話処理でI/Oエラーが発生した	処理を継続する
BA2205	IPF INPUT/OUTPUT ERROR IN SUBROUTINE SEL3	会話処理でI/Oエラーが発生した	処理を継続する
BA2206	IPF INPUT/OUTPUT ERROR IN SUBROUTINE SEL4	会話処理でI/Oエラーが発生した	処理を継続する
BA2301	DATA REPLASD TO DB FILE NORMAL END	レコードを更新した	特に無し
BA2302	PASSWORD UNMATCHED	他人のレコードの更新又は削除処理でパスワードが不一致である	正しいパスワードを入力する

8. プログラムリスト

=ANALYSIS/77= STATIC ANALYSIS OF THE FORTRAN/77 SOURCE PROGRAM DATE 1988/10/24(MONDAY) TIME 15:31:41 PAGE 0001

**
** ANALYSIS - 77 **
** (T S S - M O D E) **
**

INPUT FILE----> Q402#.AEG.FORT77
INCLUDE FILE---->
INPUT CARD ----> ELM(BAHAMA) LC(51) CO(1)TR(1)NE(1)
1. PRINT CALLING PROGRAM OFF SU(0)
2. PRINT COMMON SPECIFICATION PATTERN ON CO(1)
3. PRINT CALL/CALLED-BY REFERENCE OFF RE(0)
4. PRINT ACTUAL ARGUMENT PATTERN OFF AR(0)
5. PRINT I/O STATEMENTS OFF IO(0)
6. PRINT CHARACTER MATCHING TEST OFF MA(0)
7. PRINT PROGRAM TREE STRUCTURES ON TR(1)
8. PRINT ASSIGNMENT OF THE VALUE OFF VA(0)
9. PRINT CLASSIFIED FORTRAN STATEMENTS OFF CL(0)
10. PRINT COMMENT STATEMENTS OFF CW(0)
11. PRINT PROGRAM NEST STRUCTURES ON NE(1)

-----+
PROGRAM NAME : MAIN
THE ABOVE PROGRAM NAME IS ALREADY USED.
-----+
PROGRAM NAME : MAIN
THE ABOVE PROGRAM NAME IS ALREADY USED.
-----+

=ANALYSIS/77= LISTINGS OF THE COMMON REFERENCE TABLE DATE 1988/10/24(MONDAY) TIME 15:31:41 PAGE 0002

==== SUBPROGRAM NAME VS. COMMON BLOCK NAME ====
+-----+
|*****|AABBB.BCCDD.DDEGG.HLLLM.MMMNN.PPPRN.NSSSS.SSUVU.UUUUI
|*****|DDAAO.UHHBB.EEEEE.EAIOA.AAAEU.VAREA.AEEEE.EEADL.PSSUI
|*****|DDHIN.FAGAM.LLRTT.DBSGG.ITTNN.WGOCI.IALLL.LLDEI.DEMPI
|*****|DANO.FNDCE.DEODM.EETOP.NCCUS.WECCN.NRE12.34DLS.AREDI
|*****|BPIF.CUBCN.BTRBS.RLUUU.HH E. H . CC . ET.TCNAI
|*****|WTF.LN ET. E G. PTT. CS T. K . HT . T .ENTTI
+-----+
I 1./ADORR/ IXX . XXX.XX X . X X. XX X. XX . X XX.XXXXX.XXXXI
I 2./BITHAP/ I X . XXX.X . X. XXX. XXXI
I 3./CONDIT/ I . X. X X . X XX.XXXXX.XXXXI
I 4./ERRORH/ I X X . X. XX X. . . . X . XXXX.XXXXX.XXXXI
I 5./GEDB / I XX.XX . I
I 6./INOUT / IXX .X XXX.XX X .XXX X. XXX. XXX . XXXX.XXXXX.XXXXI
I 7./ISOU / I . . X X . X X . .X I
I 8./MANAG / I X . XXX.XX X . X X. XX X. X . X . XXX.XXXXI
I 9./NLEVEL/ I X . . X . X .XX . X . . I
I 10./PICT / I . X. . . .X X . X . XXX.XXXXX. XXXI
I 11./PROSS / I X . XXX.XX . XXX.X XX .X X . XXXX.XXXXX.XXXXI
I 12./TERN / I . X. . .X X . X . XXX.XXXXX. XXXI
I 13./UNIT / I XXX . XXX.X XXX.XXXXX.X .XX X . . XXX. XXXI
+-----+

=ANALYSIS/77= COMMON SPECIFICATION PATTERN DATE 1988/10/24(MONDAY) TIME 15:31:41 PAGE 0003

I	ADDRYR	COMMON/ADDRYR/NDBAD(32000),NITAD(16,32000),NHATCH(32000),NCON,NODM, ISEL(32000),IPRO(16),INUN(16),IS(16),IE(16),NKEYADD	ADDB	CHGDB	DBACCE	DBMENT		
I			DELDB	DELETE	GETDB	LISTUP	MAGPUT		
I			HATCHC	HATCHS	NUMSET	PROC	RECCHK		
I			SEARCH	SEL1	SEL2	SEL3	SEL4		
I			UADD	UDELETE	ULIST	UPDATE	USERCH		
I			USMENT	UUPDAT					
I	BITMAP	COMMON/BITMAP/HAPADDB	CHGDB	DBACCE	DBMENT	DELDB		
I			MAGPUT	UADD	UDELETE	ULIST	USERCH		
I			USMENT	UUPDAT					
I	CONDIT	COMMON/CONDIT/JC(3)DBMENT	DELETE	PROC	SEARCH	SEL1		
I			SEL2	SEL3	SEL4	UADD	UDELETE		
I			ULIST	UPDATE	USERCH	USMENT	UUPDAT		
I	ERRORH	COMMON/ERRORH/NMSG,NFLAG,NERCD(50),MSG(50)ADD	BAINIT	DBMENT	DELETE	ERROR		
I			GETMSG	PROC	SEARCH	SELECT	SEL1		
I			SEL2	SEL3	SEL4	UADD	UDELETE		
I			ULIST	UPDATE	USERCH	USMENT	UUPDAT		
I	GEDB	COMMON/GEDB/ISTART,IDIM,IEND,ICO,IRADSEL1	SEL2	SEL3	SEL4			
I	INOUT	COMMON/INOUT/BUFF*W(20),BUFF,COND,CSELADD	ADDB	BUFFCL	CHGDB	DBACCE		
I			DBMENT	DELDB	DELETE	GETDB	HEDER		
I			LABEL	LISTUP	MAGPUT	HATCHS	MENU		
I			NUMSET	PAGE	PROC	RECCHK	SEARCH		
I			SELECT	SEL1	SEL2	SEL3	SEL4		
I			UADD	UDELETE	ULIST	UPDATE	USERCH		
I			USMENT	UUPDAT					
I	ISOU	COMMON/ISOU/ISODELETE	PROC	SEARCH	SEL1	UPDATE		
I	MANAG	COMMON/MANAG/LRECD8,LRMAX,LRNOW,ITMAX,ITNOW,ITNAME(16),ITFORN(16), ITCHAR(16),ITEHPS,ITEMPE,IDATAS,IDATAE,SYSDADDB	CHGDB	DBACCE	DBMENT	DELDB		
I			DELETE	GETDB	LISTUP	MAGPUT	HATCHC		
I			HATCHS	NUMSET	RECCHK	SEL1	UADD		
I			UDELETE	ULIST	UPDATE	USERCH	USMENT		
I			UUPDAT						
I	MLEVEL	COMMON/MLEVEL/LEVEL(16)BAINIT	GETDB	LISTUP	HATCHC	HATCHS		

=ANALYSIS/77= COMMON SPECIFICATION PATTERN DATE 1988/10/24(MONDAY) TIME 15:31:41 PAGE 0004

I									RECCHK
I	PICT	COMMON/PICT/MIA,PIA,LSIA(32)DBMENT	MAIN	MENU	PROC	SELECT		
I			SEL1	SEL2	SEL3	SEL4	UADD		
I			UDELETE	ULIST	USERCH	USMENT	UUPDAT		
I	PROSS	COMMON/PROSS/LOGID,PDATE,IUSYOW,PASSWD,IMISS,IFUNCADD	CHGDB	DBACCE	DBMENT	DELDB		
I			DELETE	LISTUP	LOGOUT	MAGPUT	MAIR		
I			HATCHS	MENU	P*W*Y	PROC	SEARCH		
I			SELECT	SEL1	SEL2	SEL3	SEL4		
I			UADD	UDELETE	ULIST	UPDATE	USERCH		
I			USMENT	UUPDAT					
I	TERM	COMMON/TERM/IRO,IR1,TNAME,ITYPE,IRCDDBMENT	MAIN	MENU	PROC	SELECT		
I			SEL1	SEL2	SEL3	SEL4	UADD		
I			UDELETE	ULIST	USERCH	USMENT	UUPDAT		
I	UNIT	COMMON/UNIT/IDB,IADDR,ILOG,IMSG,IWORK,IList,ITSSIN,ITSSOT,IWORKY,IPWADDB	BAHAPW	BAINIT	CHGDB	DBACCE		
I			DBMENT	DELDB	ERROR	GETDB	GETMSG		
I			HEDER	LABEL	LISTUP	LOGOUT	MAGPUT		
I			MAIN	P*W*Y	PAGE	RECCHK	UADD		
I			UDELETE	ULIST	USERCH	USMENT	UUPDAT		


```

+--- TREE -----+
| FOR YOUR REFERENCE |
+-----+
| * STANDS FOR THE LIBRARY ROUTINE. |
| # STANDS FOR THE STATEMENT FUNCTION DEFINITION. |
| % STANDS FOR THE ENTRY NAME. |
| @ STANDS FOR THE EXTERNAL/INTRINSIC PROCEDURE NAME. |
| **** STANDS FOR THE ABBREVIATION OF TREE STRUCTURE. |
+-----+
    
```

USER ROUTINE

```

*****
ADD (SUBROUTINE)   ADDDB (SUBROUTINE)  BAHAPW(SUBROUTINE)  BAINIT(BLOCK DATA)  B@HOFF(SUBROUTINE)  BUFFCL(SUBROUTINE)
CHANUM(SUBROUTINE) CHGDB (SUBROUTINE)  DBACCE(SUBROUTINE)  DBMENT(SUBROUTINE)  DELDB (SUBROUTINE)  DELETE(SUBROUTINE)
ERROR (SUBROUTINE) GETDB (SUBROUTINE)  GETMSG(SUBROUTINE)  HEDER (SUBROUTINE)  LABEL (SUBROUTINE)  LISTUP(SUBROUTINE)
LOGOUT(SUBROUTINE) MAGPUT(SUBROUTINE) MAIN ( MAIN )  MATCHC(SUBROUTINE)  MATCHS(SUBROUTINE)  MENU (SUBROUTINE)
NUNSET(SUBROUTINE) P%WY (SUBROUTINE)  PAGE (SUBROUTINE)  PROC (SUBROUTINE)  RECCHK(SUBROUTINE)  MAIN ( MAIN )
MAIN ( MAIN )     SEARCH(SUBROUTINE)  SELECT(SUBROUTINE)  SEL1 (SUBROUTINE)  SEL2 (SUBROUTINE)  SEL3 (SUBROUTINE)
SELA (SUBROUTINE) UADD (SUBROUTINE)  VDELETE(SUBROUTINE) ULIST (SUBROUTINE)  UPDATE(SUBROUTINE)  USERCH(SUBROUTINE)
USMENT(SUBROUTINE) UUPDAT(SUBROUTINE)
    
```

```

MAIN ----GETMSG
+*#DATE
+*#INPRO
+*#GETJN
+*#P%WY ----BAHAPW----*NOT
+*#ERROR
+*#TRNTYP
+*#IPFCHO
+*#IPFQVS
+*#IPFQFF
+*#DBACCE----*B@HOFF----*BTEST
|
| +*#IBSET
| +*#IBCLR
+*#SELECT----*BUFFCL
|
| +*#IPFHIO
| +*#ERROR
| +*#PROC ----*BUFFCL
|
| +*#IPFHIO
| +*#ERROR
| +*#ADD ----*COHAND
|
| +*#ERROR
| +*#MATCHS----*CHANUM----*LENGTH
|
| +*#MATCHC
|
| +*#GETDB ----*ABS
|
| +*#ADDB ----*CHANUM----*LENGTH
|
| +*#B@HOFF----*BTEST
|
| +*#IBSEY
|
| +*#IBCLR
|
| +*#MAGPUT
+*#SEARCH----*COHAND
|
| +*#MATCHS----*CHANUM----*LENGTH
|
| +*#MATCHC
|
| +*#ERROR
|
| +*#SEL1 ----*GETDB ----*ABS
|
| +*#ERROR
|
| +*#IPFHIO
|
| +*#SEL2 ----*GETDB ----*ABS
|
| +*#IPFHIO
|
| +*#ERROR
|
| +*#SEL3 ----*GETDB ----*ABS
|
| +*#IPFHIO
|
| +*#ERROR
|
| +*#SEL4 ----*GETDB ----*ABS
|
| +*#IPFHIO
|
| +*#ERROR
|
| +*#LISTUP----*MDD
|
| +*#HEDER
|
| +*#PAGE
|
| +*#LABEL
    
```

```

I          I          I          I          +--LISTUP----*MOD
I          I          I          I          +--HEDER
I          I          I          I          +--PAGE
I          I          I          I          +--LABEL
I          I          I          I          +--LISTUP----*MOD
I          I          I          I          +--HEDER
I          I          I          I          +--PAGE
I          I          I          I          +--LABEL
I          I          I          I          +--LISTUP----*MOD
    
```

-ANALYSIS/77- ** TREE STRUCTURE ** ENTRY POINT = MAIN DATE 1988/10/24(MONDAY) TIME 15:31:41 PAGE 0007

```

MAIN  ----SELECT----PROC  ----SEARCH----SEL1  ----LISTUP----HEDER
I          I          I          I          +--PAGE
I          I          I          I          +--LABEL
I          I          I          I          +--LISTUP----*MOD
I          I          I          I          +--HEDER
I          I          I          I          +--PAGE
I          I          I          I          +--LABEL
I          I          I          I          +--UPDATE----CHGDB  ----CHANUM----*LENGTH
I          I          I          I          +--LOGOUT
I          I          I          I          +--ERROR
I          I          I          I          +--SEARCH----*COMAND
I          I          I          I          +--MATCHS----CHANUM----*LENGTH
I          I          I          I          +--MATCHC
I          I          I          I          +--ERROR
I          I          I          I          +--SEL1  ----GETDB  ----*ABS
I          I          I          I          +--ERROR
I          I          I          I          +--*IPFNIO
I          I          I          I          +--SEL2  ----GETDB  ----*ABS
I          I          I          I          +--*IPFNIO
I          I          I          I          +--ERROR
I          I          I          I          +--SEL3  ----GETDB  ----*ABS
I          I          I          I          +--*IPFNIO
I          I          I          I          +--ERROR
I          I          I          I          +--SEL4  ----GETDB  ----*ABS
I          I          I          I          +--*IPFNIO
I          I          I          I          +--ERROR
I          I          I          I          +--LISTUP----*MOD
I          I          I          I          +--HEDER ****
I          I          I          I          +--PAGE ****
I          I          I          I          +--LABEL ****
I          I          I          I          +--LISTUP----*MOD
I          I          I          I          +--HEDER
I          I          I          I          +--PAGE
I          I          I          I          +--LABEL
I          I          I          I          +--LISTUP----*MOD
I          I          I          I          +--HEDER
I          I          I          I          +--PAGE
I          I          I          I          +--LABEL
I          I          I          I          +--LISTUP----*MOD
I          I          I          I          +--HEDER
I          I          I          I          +--PAGE
I          I          I          I          +--LABEL
I          I          I          I          +--GETDB  ----*ABS
I          I          I          I          +--*PROMPT
I          I          I          I          +--*BANAPV----*NOT
I          I          I          I          +--DELETE----SEARCH----*COMAND
I          I          I          I          +--MATCHS----CHANUM----*LENGTHII
I          I          I          I          +--MATCHC
I          I          I          I          +--ERROR
I          I          I          I          +--SEL1  ----GETDB  ----*ABS
I          I          I          I          +--ERROR
I          I          I          I          +--*IPFNIO
I          I          I          I          +--SEL2  ----GETDB  ----*ABS
    
```

```

I          I          I          +-*IPFNIO
I          I          I          +-*ERROR
I          I          I          +-*SEL3 -----GETDD ----*AOS
I          I          I          +-*IPFNIO
I          I          I          +-*ERROR
I          I          I          +-*SEL4 -----GETDB ----*ABS
I          I          I          +-*IPFNIO
I          I          I          +-*ERROR
I          I          I          +-*LISTUP----*MOD
    
```

*ANALYSIS/77= ** TREE STRUCTURE ** ENTRY POINT = MAIN DATE 1988/10/24(MONDAY) TIME 15:31:41 PAGE 0008

```

MAIN -----SELECT-----PROC -----DELETE-----SEARCH-----SEL1 -----SEL2 -----SEL3 -----SEL4 -----LISTUP-----+-*HEADER ----
I          I          I          I          I          I          I          I          I          +-*PAGE ----
I          I          I          I          I          I          I          I          I          +-*LABEL ----
I          I          I          I          I          I          I          I          I          +-*LISTUP----*MOD
I          I          I          I          I          I          I          I          I          +-*HEADER
I          I          I          I          I          I          I          I          I          +-*PAGE
I          I          I          I          I          I          I          I          I          +-*LABEL
I          I          I          I          I          I          I          I          I          +-*LISTUP----*MOD
I          I          I          I          I          I          I          I          I          +-*HEADER
I          I          I          I          I          I          I          I          I          +-*PAGE
I          I          I          I          I          I          I          I          I          +-*LABEL
I          I          I          I          I          I          I          I          I          +-*LISTUP----*MOD
I          I          I          I          I          I          I          I          I          +-*HEADER
I          I          I          I          I          I          I          I          I          +-*PAGE
I          I          I          I          I          I          I          I          I          +-*LABEL
I          I          I          I          I          I          I          I          I          +-*GETDB ----*ABS
I          I          I          I          I          I          I          I          I          +-*PROMPT
I          I          I          I          I          I          I          I          I          +-*BAHAPV----*HOT
I          I          I          I          I          I          I          I          I          +-*DELOD ----*ONOFF----*BTEST
I          I          I          I          I          I          I          I          I          I          +-*ISSET
I          I          I          I          I          I          I          I          I          I          +-*IBCLR
I          I          I          I          I          I          I          I          I          I          +-*HAGPUT
I          I          I          I          I          I          I          I          I          I          +-*LOGOUT
+-*MENU -----*IPFNIO
+-*ERROR
+-*OSMENT-----*IPFNIO
I          I          I          +-*ERROR
I          I          I          +-*UADD -----ERROR
I          I          I          I          +-*COHAND
I          I          I          I          +-*HOT
I          I          I          +-*UUPDAT-----ERROR
I          I          I          I          +-*COHAND
I          I          I          I          +-*HOT
I          I          I          +-*USERCH-----ERROR
I          I          I          I          +-*COHAND
I          I          I          I          +-*HOT
I          I          I          +-*UDELETE-----ERROR
I          I          I          I          +-*COHAND
I          I          I          I          +-*HOT
+-*DBMENT-----*IPFNIO
+-*ERROR
+-*COHAND
+-*ULIST -----*COHAND
I          I          I          +-*HOT
+-*IPFCVS
    
```

```

=====
I   T H E   C L A S S I F I C A T I O N   O F   F O R T R A N   S T A T E M E N T S
=====
I  COMMENT                2046( 45.90 %)      X      WAIT                0( 0.0 %)      I
I  CONTINUATION           295(  6.62 %)      X      INQUIRE               0( 0.0 %)      I
I  ASSIGNMENT             445(  9.98 %)      X      COMPLEX                0( 0.0 %)      I
I  ASSIGN                 0(  0.0 %)      X      LOGICAL               36(  0.81 %)   I
I  GO TO                  100(  2.24 %)     X      INTEGER              163(  3.66 %)   I
I  ASSIGNED GO TO        0(  0.0 %)      X      CHARACTER             164(  3.68 %)   I
I  COMPUTED GO TO        1(  0.02 %)      X      DOUBLE PRECISION      0(  0.0 %)      I
I  ARITHMETIC IF         0(  0.0 %)      X      REAL                  0(  0.0 %)      I
I  LOGICAL IF            104(  2.33 %)     X      EQUIVALENCE           8(  0.18 %)     I
I  ASSIGNMENT            38 PRINT                0 X      DATA                 18(  0.40 %)     I
I  ASSIGN                0 PUNCH                0 X      NAMELIST              0(  0.0 %)      I
I  GO TO                 49 REWIND              0 X      IMPLICIT              0(  0.0 %)      I
I  ASSIGNED GO TO        0 FIND                 0 X      PARAMETER             0(  0.0 %)      I
I  COMPUTED GO TO        0 WAIT                0 X      COMMON                224(  5.02 %)   I
I  ARITHMETIC IF         0 CALL                15 X     SAVE                  0(  0.0 %)      I
I  OPEN                  0 RETURN              2 X      DIMENSION             0(  0.0 %)      I
I  CLOSE                 0 STOP                0 X      EXTERNAL              0(  0.0 %)      I
I  READ                  0 PAUSE               0 X      INTRINSIC             0(  0.0 %)      I
I  WRITE                 0 DECODE              0 X      DEFINE FILE           0(  0.0 %)      I
I  BACKSPACE             0 ENCODE              0 X      PROGRAM               0(  0.0 %)      I
I  ENDFILE               0 INQUIRE            0 X      BLOCKDATA             1(  0.02 %)     I
I  IF (...) THEN        138(  3.10 %)     X      FUNCTION              0(  0.0 %)      I
I  DO                    60(  1.35 %)     X      SUBROUTINE            40(  0.90 %)     I
I  DO UNTIL(...)        0(  0.0 %)      X      ENTRY                 0(  0.0 %)      I
I  DO WHILE(...)        0(  0.0 %)      X      CALL                  125(  2.80 %)   I
I  ELSE                  24(  0.54 %)     X      RETURN               44(  0.99 %)     I
I  ELSE IF               4(  0.09 %)      X      STOP                  3(  0.07 %)     I
I  END IF                138(  3.10 %)     X      PAUSE                 0(  0.0 %)      I
I  FORMAT                13(  0.29 %)     X      END                   44(  0.99 %)     I
I  OPEN                  8(  0.18 %)      X      CONTINUE             110(  2.47 %)   I
I  CLOSE                 9(  0.20 %)      X      DECODE               0(  0.0 %)      I
I  READ                  37(  0.83 %)     X      ENCODE               0(  0.0 %)      I
I  WRITE                 46(  1.03 %)     X      DEBUG                0(  0.0 %)      I
I  BACKSPACE             1(  0.02 %)      X      AT                    0(  0.0 %)      I
I  ENDFILE               0(  0.0 %)      X      DISPLAY              0(  0.0 %)      I
I  PRINT                 0(  0.0 %)      X      INIT                  0(  0.0 %)      I
I  PUNCH                 0(  0.0 %)      X      TRACE                 0(  0.0 %)      I
I  REWIND                6(  0.13 %)     X      -- ROGUE --          0(  0.0 %)      I
I  FIND                  0(  0.0 %)      X      NCHARACTER (JEF)     3(  0.07 %)     I
=====

```

TOTAL STATEMENTS = 4458

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C  SUBROUTINE ADD                                *
C*****
SUBROUTINE ADD
C
C-----
C-  PROCESS NOW CONDITION                       -
C-----
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUS*OW, IMISS, IFUNC
C-----
C-  ADDR FILES DATA                           -
C-----
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+                NMATCH(32000),NCON, NDOM,
+                ISEL(32000), IPRD(16), INUH(16),
+                IS(16), IE(16),NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUH, IS, IE, NKEY
LOGICAL*1 ISEL, IPRD
C-----
C-  FILE INPUT/OUTPUT BUFFER                   -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF=1312, COND=384, CSEL=384
C-----
C-  ERROR MESSAGE AREA                         -
C-----
COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)
C
CHARACTER MFLAG*2, MSG*60
INTEGER*2 NMSG, NERCD
C
C = = COND AREA CLEAR
DO 100 I = 0 , 360 , 24
COND(I+1:I+24) = 24H
100 CONTINUE
C
C = = TSS COMMAND CHECK
IF( BUFF(225:249) .NE. 25H .OR.
+  BUFF(250:274) .NE. 25H ) THEN
CALL COMAND (BUFF(225:274))
BUFF(225:249) = 25H
BUFF(250:274) = 25H
GOTO 999
ENDIF
C = = DATA SET NAME CHECK
IF( BUFF(11:32) .EQ. 22H .AND.
+  BUFF(33:54) .EQ. 22H ) THEN

```

DATA SET NAME : '0402#.AEG.FORT77(BAHAMA)'

.....1.....2.....3.....4.....5.....6.....7.....8

```

CALL ERROR ( 2105 )
GOTO 999
ENDIF
C
C = = FILE NAME DOUBLE CHECK
COND( IS(1):IE(1) ) = BUFF(11:54)
CALL MATCHS
CALL MATCHC
C**** VERSION UP ****C
**** IF( IUS*OW .NE. 2 ) CALL RECCHK
C**** VERSION UP ****C
ISTART = 1
IDIM = +1
CALL GETDB ( ISTART , IDIM , IEND , ICO , IRAD , IER )
IF( ICO .GT. 0 ) THEN
CALL ERROR ( 2101 )
GOTO 999
ENDIF
C
C = = DATA INPUT TO DATA BASE
COND( IS(1):IE(1) ) = BUFF(11:54)
COND( IS(2):IE(2) ) = BUFF(55:74)
COND( IS(3):IE(3) ) = LOGID(1:5)
COND( IS(4):IE(4) ) = PDATE
I1 = 7
DO 200 I = 74 , 154 , 20
IF( BUFF(I+1:I+20) .NE. 20H ) THEN
COND( IS(11):IE(11) ) = BUFF(I+1:I+20)
I1 = I1 + 1
ENDIF
200 CONTINUE
COND( IS(12):IE(12) ) = BUFF(175:224)
C
C = = TO ADD THE ADDDB
CALL ADDDB ( IER )
C
IF( IER .EQ. 1 ) CALL ERROR ( 2102 )
C
IF( IER .EQ. 2 ) CALL ERROR ( 2103 )
C
IF( IER .NE. 1 .AND. IER .NE. 2 ) CALL ERROR ( 2104 )
C
999 RETURN
END

```

DATE : 88/10/25

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C= SUBROUTINE ADDDB = 00010000
C***** 00010100
SUBROUTINE ADDDB ( IER ) 00010200
C 00010300
C 00010400
C----- 00010500
C= FUNCTION = 00010600
C= TO ADD THE DB RECORD AND ADDR RECORD = 00010700
C= PARAMETER = 00010800
C= IER : ERROR RETURN CODE (0) = 00010900
C----- 00011000
C 00011100
C----- 00011200
C= DATABASE MANAGEMENT AREA - 00011300
C----- 00011400
COMMON / MANAG / LRECD, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID 00011500
C 00011600
CHARACTER ITNAME*24, ITFORM*24, SYSID*5 00011700
INTEGER*2 LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE 00011800
C 00011900
C----- 00012000
C= ADDR FILES DATA - 00012100
C----- 00012200
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRO(16), INUH(16),
+ IS(16), IE(16), NKEY 00012300
C 00012400
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUH, IS, IE, NKEY 00012500
LOGICAL*1 ISEL, IPRO 00012600
C----- 00012700
C= UNIT INPUT OUTPUT NUMBER - 00012800
C----- 00012900
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*, IPW 00013000
C 00013100
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*, IPW 00013200
C----- 00013300
C= FILE INPUT/OUTPUT BUFFER - 00013400
C----- 00013500
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL 00013600
C 00013700
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384 00013800
C----- 00013900
C= DB FILE EMPTY RECORD CONTROL - 00014000
C----- 00014100
COMMON / BITMAP / MAP 00014200
C 00014300
C----- 00014400
C----- 00014500
C----- 00014600
C----- 00014700
C----- 00014800
C----- 00014900
C----- 00015000
C= DB FILE EMPTY RECORD CONTROL - 00015100
C----- 00015200
COMMON / BITMAP / MAP 00015300
C 00015400

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C CHARACTER MAP*4000 00015500
C IER = 0 00015600
C IF ( LRNOW .GE. LRMAX+IDATAS-1 ) THEN 00015700
IER = 1 00015800
GO TO 999 00015900
END IF 00016000
C LRNOW = LRNOW + 1 00016100
C= TO CHENGE CHARACTER TO NUMBER 00016200
C= 00016300
DO 100 I=1,ITNOW 00016400
CALL CHANUM ( COND(IS(I):IE(I)) , INUH(I) ) 00016500
NITAD(I,LRNOW) = INUH(I) 00016600
CDS=> 00016700
* WRITE(6,*) NITAD(I,LRNOW),INUH(I) 00016800
CDE=> 00016900
100 CONTINUE 00017000
C= CONTROL BIT ON & RECORD PUT & PRG AREA UPDATE 00017100
C= 00017200
LM = LRMAX 00017300
CALL BONDOFF ( MAP , 1 , LM , 0 , IB , 1 , IERR ) 00017400
IF ( IERR .NE. 0 ) THEN 00017500
IER = 2 00017600
GO TO 999 00017700
END IF 00017800
C ISEL (LRNOW) = .FALSE. 00017900
NMATCH(LRNOW) = 0 00018000
NDBAD (LRNOW) = IB + IDATAS - 1 00018100
CDS * WRITE(6,*) ' IB= ',IB,' IDATAS= ',IDATAS,' NDBAD(LRNOW)= ',
+ NDBAD(LRNOW) 00018200
CDE 00018300
C= BIT MAP RECORD PUT 00018400
C= 00018500
CALL MAGPUT ( NDBAD(LRNOW) , IR ) 00018600
C= USER DATA RECORD PUT 00018700
C= 00018800
WRITE( IDB , '(A)' , REC=NDBAD(LRNOW) ) COND 00018900
C= ADDR DATA RECORD PUT 00019000
C= 00019100
WRITE( IADDR , REC=NDBAD(LRNOW) )
+ NDBAD(LRNOW) , (NITAD(L,LRNOW),L=1,ITMAX) 00019200
C 999 RETURN 00019300
END 00019400
00019500
00019600
00019700
00019800
00019900
00020000
00020100
00020200
00020300
00020400
00020500
00020600
00020700

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*****
C*  SUBROUTINE BAHAPW *
C*****
SUBROUTINE BAHAPW ( LOGID , UPW , OPW )
C
C-----
C=  FUNCTION =
C=  TO GENERATE THE USER PASSWRD. =
C=  PARAMETER =
C=  UID ; LOGON USER ID (I) =
C=  UPW ; USER PASSWORD (O) =
C=  OPW ; OWNER PASSWORD (O) =
C-----
C
C-----
C-  UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*, IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*, IPW
C
CHARACTER LOGID*8
C
CHARACTER CUP(2)*4 , CSP(2)*4 , CDP(2)*4
C
INTEGER IUP(2) , ISP(2) , IDP(2)
C
EQUIVALENCE (CUP,IUP) , (CSP,ISP) , (CDP,IDP)
C
CHARACTER UPW*8 , OPW*8
C
UPW = 8H
OPW = 8H
C
100 CONTINUE
READ ( IPW , '(6A4)' , END = 888 )
+ CDP(1),CDP(2),CUP(1),CUP(2),CSP(1),CSP(2)
IDP(1) = NOT(IDP(1))
IDP(2) = NOT(IDP(2))
C
IF ( LOGID(1:4).NE.CDP(1)(1:4).OR.LOGID(5:5).NE.CDP(2)(1:1) )
+ GO TO 100
C
IUP(1) = NOT(IUP(1))
IUP(2) = NOT(IUP(2))
ISP(1) = NOT(ISP(1))
ISP(2) = NOT(ISP(2))
C
UPW = CUP(1)//CUP(2)
OPW = CSP(1)//CSP(2)
CCCC GO TO 999
C
888 CONTINUE

```

```

00020800
00020900
00021000
00021100
00021200
00021300
00021400
00021500
00021600
00021700
00021800
00021900
00022000
00022100
00022200
00022300
00022400
00022500
00022600
00022700
00022800
00022900
00023000
00023100
00023200
00023300
00023400
00023500
00023600
00023700
00023800
00023900
00024000
00024100
00024200
00024300
00024400
00024500
00024600
00024700
00024800
00024900
00025000
00025100
00025200
00025300
00025400
00025500
00025600
00025700
00025800
00025900
00026000
00026100
00026200

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C  REWIND IPW 00026300
C 999 CONTINUE 00026400
C RETURN 00026500
C END 00026600
00026700

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* BLOCKDATA BAINIT *
C*****
BLOCKDATA BAINIT
C
C-----
C= FUNCTION =
C= TO SET THE COMMON CONSTANT DATA =
C-----
C
C-----
C- ERROR MESSAGE AREA -
C-----
COMMON / ERRORM / NMSG, HFLAG, NERCD(50), MSG(50)
C
CHARACTER HFLAG*2, MSG*60
INTEGER*2 NMSG, NERCD
C
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*, IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*, IPW
C
C-----
C- MATCH LEVEL CHECK KEY -
C-----
COMMON / MLEVEL / LEVEL(16)
C
LOGICAL*1 LEVEL
C
DATA HFLAG /'BA'/
C
DATA IDB /10/, IADDR/20/, ILOG /30/, IMSG /40/,
+ IWORK/50/, ILIST/60/, ITSSIN/05/, ITSSOT/06/, IWORK*/70/
DATA IPW /90/
C
DATA LEVEL / 2*.FALSE., 9*.TRUE., 5*.FALSE. /
C
END
    
```

```

00026800
00026900
00027000
00027100
00027200
00027300
00027400
00027500
00027600
00027700
00027800
00027900
00028000
00028100
00028200
00028300
00028400
00028500
00028600
00028700
00028800
00028900
00029000
00029100
00029200
00029300
00029400
00029500
00029600
00029700
00029800
00029900
00030000
00030100
00030200
00030300
00030400
00030500
00030600
00030700
00030800
00030900
00031000
00031100
    
```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE BONOFF *
C*****
SUBROUTINE BONOFF ( IAREA , ISB , IEB , ION , IB , IMODE , IER )
C
C-----
C= FUNCTION =
C= TO BIT TEST THE IAREA =
C= PARAMETER =
C= IAREA ; BIT CHECK AREA =
C= ISB ; CHECK START BIT =
C= IEB ; CHECK END BIT =
C= ION ; CHECK FLAG =
C= 0 : BIT OFF CHECK =
C= 1 : BIT ON CHECK =
C= IB ; BIT LOCATION FROM 1 BIT =
C= IMODE ; PROCESS MODE =
C= 0 : NON PROCESS =
C= 1 : BIT RIVERS =
C= IER ; ERROR RETURN CODE =
C= -999: NOTHING OF BIT ON OR OFF =
C-----
C
INTEGER IAREA(1)
LOGICAL TF
IER = 0
IB = 0
C
DO 100 I=ISB,IEB
IWORD = (I-1) * 0.03125 + 1
IBIT = -(I-(IWORD-1)*32) + 33
TF = BTEST ( IAREA(IWORD) , IBIT-1 )
C
IF ( ( ION.EQ.0 ) .AND. ( .NOT.TF ) ) .OR.
+ ( ION.EQ.1 ) .AND. ( TF ) ) THEN
IB = 1
IF ( IMODE .EQ. 1 .AND. ION .EQ. 0 )
+ IAREA(IWORD) = IBSET(IAREA(IWORD),IBIT-1)
+ IF ( IMODE .EQ. 1 .AND. ION .EQ. 1 )
+ IAREA(IWORD) = IBCLR(IAREA(IWORD),IBIT-1)
GO TO 999
END IF
C
100 CONTINUE
IER = -999
C
999 RETURN
END
    
```

```

00031200
00031300
00031400
00031500
00031600
00031700
00031800
00031900
00032000
00032100
00032200
00032300
00032400
00032500
00032600
00032700
00032800
00032900
00033000
00033100
00033200
00033300
00033400
00033500
00033600
00033700
00033800
00033900
00034000
00034100
00034200
00034300
00034400
00034500
00034600
00034700
00034800
00034900
00035000
00035100
00035200
00035300
00035400
00035500
00035600
00035700
00035800
00035900
    
```

- 77 -

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C*  SUBROUTINE BUFFCL *
C*****
      SUBROUTINE BUFFCL
C
C=====
C=  FUNCTION =
C=    TO CLEAR THE BUFFER AREA =
C=====
C-----
C-  FILE INPUT/OUTPUT BUFFER -
C-----
      COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
      CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C
      DO 100 I=1,18
      DO 100 J=0,320,64
      BUFF*W(I)(J+1:J+64) = 64H
      +
100 CONTINUE
C
      DO 200 I=0,1200,100
      BUFF(I+1:I+100) = 100H
      +
200 CONTINUE
C
      BUFF(1301:1312) = 12H
C
      DO 300 I=0,320,64
      COND(I+1:I+64) = 64H
      +
300 CONTINUE
      RETURN
      END

```

```

00036000
00036100
00036200
00036300
00036400
00036500
00036600
00036700
00036800
00036900
00037000
00037100
00037200
00037300
00037400
00037500
00037600
00037700
00037800
00037900
00038000
00038100
00038200
00038300
00038400
00038500
00038600
00038700
00038800
00038900
00039000
00039100
00039200
00039300
00039400

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C*  SUBROUTINE CHANUM *
C*****
      SUBROUTINE CHANUM ( A , NUM )
C
C=====
C=  FUNCTION =
C=    TO CHANGE THE CHARACTER TO NUMBER =
C=====
C-----
C-  PARAMETER =
C=    A : CHARACTER (I) =
C=    NUM : CHANGED NUMBER (O) =
C-----
C
      CHARACTER A*(*),WORK*2
      INTEGER*2 IWORK,NUM
      EQUIVALENCE (WORK,IWORK)
C
      NUM = 0
      IWORK = 0
      CALL LENGTH(A,IN)
      IF ( IN .EQ. 0 ) RETURN
      DO 100 I=1,IN
      WORK(2:2) = A(I:I)
100 NUM=NUM+IWORK
C
      RETURN
      END

```

```

00039500
00039600
00039700
00039800
00039900
00040000
00040100
00040200
00040300
00040400
00040500
00040600
00040700
00040800
00040900
00041000
00041100
00041200
00041300
00041400
00041500
00041600
00041700
00041800
00041900
00042000
00042100

```

- 78 -

PNC N9520 88-022(2)

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C*  SUBROUTINE CHGDB                *
C*****
SUBROUTINE CHGDB ( IER )
C
C-----
C=  FUNCTION                        =
C=    TO CHANGE THE DB RECORD AND ADDR RECORD  =
C=  PARAMETER                      =
C=    IER : ERROR RETURN CODE (0)           =
C-----
C-  PROCCSS NOW CONDITION          -
C-----
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUS*OW, IMISS, IFUNC
C-----
C-  DATABASE MANAGEMENT AREA      -
C-----
COMMON / MANAG / LRECOB, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LRECOB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE
C-----
C-  ADDR FILES DATA              -
C-----
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRD(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRD
C-----
C-  UNIT INPUT OUTPUT NUMBER      -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORKY, IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORKY, IPW
C-----
C-  FILE INPUT/OUTPUT BUFFER      -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL

```

```

00042200
00042300
00042400
00042500
00042600
00042700
00042800
00042900
00043000
00043100
00043200
00043300
00043400
00043500
00043600
00043700
00043800
00043900
00044000
00044100
00044200
00044300
00044400
00044500
00044600
00044700
00044800
00044900
00045000
00045100
00045200
00045300
00045400
00045500
00045600
00045700
00045800
00045900
00046000
00046100
00046200
00046300
00046400
00046500
00046600
00046700
00046800
00046900
00047000
00047100
00047200
00047300
00047400
00047500
00047600

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C-----
C-  DB FILE EMPTY RECORD CONTROL  -
C-----
COMMON / BITMAP / MAP
C
CHARACTER MAP*4000
C
IER = 0
C=
C=  TO SET THE PGW AREA , DB & ADDR RECORD
C=
DO 200 I=1,LRNOW
IF ( ISEL(I) ) THEN
DO 100 J=1,ITNOW
CALL CHANUM ( COND(IS(J):IE(J)) , INUM(J) )
NITAD(J,I) = INUM(J)
100 CONTINUE
C=
C=  TO PUT THE DB RECORD
C=
WRITE( IDB , '(A)' , REC=NDBAD(I) ) COND
C=  TO PUT THE LOG RECORD
C=
CALL LOGOUT ( 2 , CSEL , IERCD )
C=  TO PUT THE ADDR RECORD
C=
WRITE( IADDR , REC=NDBAD(I) )
+ NDBAD(I), (NITAD(LL,I), LL=1, ITMAX)
C
GO TO 999
END IF
200 CONTINUE
C
IER = 1
C
999 RETURN
END

```

```

00047700
00047800
00047900
00048000
00048100
00048200
00048300
00048400
00048500
00048600
00048700
00048800
00048900
00049000
00049100
00049200
00049300
00049400
00049500
00049600
00049700
00049800
00049900
00050000
00050100
00050200
00050300
00050400
00050500
00050600
00050700
00050800
00050900
00051000
00051100
00051200
00051300
00051400
00051500
00051600
00051700
00051800

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*****
C* SUBROUTINE DBOPEN&CLOSE *
C*****
SUBROUTINE DBACCE ( IOP , IERR )
C
C=====
C= FUNCTION =
C= TO OPEN DATA BESE & SET THE DBMA AREA =
C= PARAMETER =
C= IOP ; OPERATION FLAG (1) =
C= 1 : DB OPEN & SET THE MANAG. AREA =
C= 2 : DB CLOSE =
C= IER ; ERROR CODE (0) =
C=====
C- PROCCES NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUS*OW, IMISS, IFUNC
C-----
C- DATABASE MANAGEMENT AREA -
C-----
COMMON / MANAG / LRECCB, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LRECCB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*, IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*, IPW
C-----
C- ADDR FILES DATA -
C-----
COMMON / ADDR*YR / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C-----
00051900
00052000
00052100
00052200
00052300
00052400
00052500
00052600
00052700
00052800
00052900
00053000
00053100
00053200
00053300
00053400
00053500
00053600
00053700
00053800
00053900
00054000
00054100
00054200
00054300
00054400
00054500
00054600
00054700
00054800
00054900
00055000
00055100
00055200
00055300
00055400
00055500
00055600
00055700
00055800
00055900
00056000
00056100
00056200
00056300
00056400
00056500
00056600
00056700
00056800
00056900
00057000
00057100
00057200
00057300

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C- DB FILE EMPTY RECORD CONTROL -
C-----
COMMON / BITHAP / MAP
C
CHARACTER MAP*4000
C-----
C- FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF=1312, COND=384, CSEL=384
C
IER = 0
C
GO TO ( 100 , 1000 ) , IOP
RETURN
C
100 CONTINUE
C
READ ( IDB , '(35X,A5)' ) SYSID
CLOSE (IDB)
C
C= DB FILE OPEN
C=
OPEN ( UNIT=IDB, FILE=SYSID/'BAHAMA.DB', STATUS='OLD',
+ ACCESS='DIRECT', FORM='FORMATTED', RECL=384)
C= ADDR FILE OPEN
C=
OPEN ( UNIT=IADDR, FILE=SYSID/'BAHAMA.ADDR', STATUS='OLD',
+ ACCESS='DIRECT', FORM='FORMATTED', RECL=34)
C
READ ( IDB , '(3I5,2I2,3I5)' , REC=1 ) LRECCB, LRMAX,
+ LRNOW, ITMAX, ITNOW, ITEMPS, ITEMPE, IDATAS
EDS
* WRITE(6,*) 'DBACCE LRNOW,ITNOW ',LRNOW,ITNOW
CDE
C
READ ( IDB , '(100A24)' , REC=2 ) (ITNAME(J),J=1,ITMAX)
C
READ ( IDB , '(100A24)' , REC=3 ) (ITFORM(J),J=1,ITMAX)
C
READ ( IDB , '(100I3)' , REC=4 ) (ITCHAR(J),J=1,ITMAX)
C
IDATAE = IDATAS + LRMAX
C
DO 500 J=ITEMPS,ITEMPE-1
READ ( IDB , '(A)' , REC=J ) BUFF*W(20)
I1 = LRECCB * (J-ITEMPS) + 1
I2 = I1 + LRECCB - 1
MAP(I1:I2) = BUFF*W(20)
500 CONTINUE
C
00057400
00057500
00057600
00057700
00057800
00057900
00058000
00058100
00058200
00058300
00058400
00058500
00058600
00058700
00058800
00058900
00059000
00059100
00059200
00059300
00059400
00059500
00059600
00059700
00059800
00059900
00060000
00060100
00060200
00060300
00060400
00060500
00060600
00060700
00060800
00060900
00061000
00061100
00061200
00061300
00061400
00061500
00061600
00061700
00061800
00061900
00062000
00062100
00062200
00062300
00062400
00062500
00062600
00062700
00062800

```

- 80 -

```

+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8
READ ( IDB , 'A' , REC=ITEMPE ) BUFFW(20)          00062900
I1 = LRECD * (ITEMPE-ITEMPS) + 1                    00063000
I2 = LRMAX / 8                                       00063100
MAP(I1:I2) = BUFFW(20)                               00063200
C                                                     00063300
I1 = 0                                                00063400
DO 900 I=1,LRMAX                                     00063500
  CALL BONOFF ( MAP, I, I, 1, IB, 0, IER )           00063600
  IF ( IB .EQ. I ) THEN                              00063700
    I1 = I1 + 1                                      00063800
    IA = I + IDATAS - 1                             00063900
    READ ( IADDR, REC=IA ) NDBAD(I1), (NITAD(J,I1),J=1,ITMAX) 00064000
CDS                                                  00064100
*   WRITE(6,*) ' DBACCE NDBAD,NITAD ',              00064200
* + NDBAD(I1),(NITAD(K1,I1),K1=1,ITMAX)             00064300
CDE                                                  00064400
  IF ( I1 .GE. LRNOW ) GO TO 930                    00064500
  END IF                                             00064600
900 CONTINUE                                         00064700
C                                                     00064800
930 CONTINUE                                         00064900
  IEE = 0                                            00065000
  DO 950 I=1,ITNOW                                   00065100
    ISS = IEE + 1                                    00065200
    IEE = ISS + ITCHAR(I) - 1                       00065300
    IS(I) = ISS                                      00065400
    IE(I) = IEE                                      00065500
950 CONTINUE                                         00065600
  RETURN                                             00065700
C                                                     00065800
1000 CONTINUE                                        00065900
  CLOSE ( IDB )                                     00066000
  CLOSE ( IADDR)                                   00066100
C                                                     00066200
  RETURN                                             00066300
  END                                               00066400

```

```

+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8
SUBROUTINE DBMHT                                     00066500
C                                                     00066600
C-----                                           00066700
C= FUNCTION                                         00066800
C= TO DISPLAY THE OWNER DB MENTE. PICTURE =       00066900
C-----                                           00067000
C                                                     00067100
C-----                                           00067200
C- PROCCES NOW CONDITION                           00067300
C-----                                           00067400
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC 00067500
C                                                     00067600
CHARACTER LOGID*8, PDATE*8, PASSWD*4              00067700
INTEGER*2 IUS*OW, IMISS, IFUNC                    00067800
C                                                     00067900
C-----                                           00068000
C- DATABASE MANAGEMENT AREA                        00068100
C-----                                           00068200
COMMON / MANAG / LRECD, LRMAX, LRNOW, ITMAX, ITNOW, 00068300
+ ITNAME(16), ITFORM(16), ITCHAR(16),            00068400
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID          00068500
C                                                     00068600
CHARACTER ITNAME*24, ITFORM*24, SYSID*5           00068700
INTEGER*2 LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS, 00068800
+ ITEMPE, IDATAS, IDATAE                         00068900
C-----                                           00069000
C- PROCESS CONDITION AREA                          00069100
C-----                                           00069200
COMMON / CONDIT / JC(3)                           00069300
C                                                     00069400
INTEGER*2 JC                                       00069500
C-----                                           00069600
C- ERROR MESSAGE AREA                              00069700
C-----                                           00069800
COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50) 00069900
C                                                     00070000
CHARACTER MFLAG*2, MSG*60                          00070100
INTEGER*2 NMSG, NERCD                              00070200
C-----                                           00070300
C- UNIT INPUT OUTPUT NUMBER                        00070400
C-----                                           00070500
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST, 00070600
+ ITSSIN, ITSSOT, IWORK*, IPW                    00070700
C                                                     00070800
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT, 00070900
+ IWORK*, IPW                                     00071000
C-----                                           00071100
C- ADDR FILES DATA                                00071200
C-----                                           00071300
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000), 00071400
+ NMATCH(32000), NCON, NDOH,                      00071500

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
+
+ ISEL(32000), IPRO(16), INUH(16), 00072000
+ IS(16), IE(16), NKEY 00072100
C INTEGER=2 HD9AD, NITAD, NMATCH, NCON, NDOM, INUH, IS, IE, NKEY 00072200
LOGICAL=1 ISEL, IPRO 00072300
C 00072400
C 00072500
C----- 00072600
C- DB FILE EMPTY RECORD CONTROL - 00072700
C----- 00072800
COMMON / BITMAP / MAP 00072900
C 00073000
CHARACTER MAP=4000 00073100
C 00073200
C----- 00073300
C- TERMTYPE COMMON AREA - 00073400
C----- 00073500
COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD 00073600
C 00073700
CHARACTER TNAME=8 00073800
C 00073900
C----- 00074000
C- FILE INPUT/OUTPUT BUFFER - 00074100
C----- 00074200
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL 00074300
C 00074400
CHARACTER BUFF*W=384, BUFF=1312, COND=384, CSEL=384 00074500
C 00074600
C----- 00074700
C- PICTURE COMMON AREA - 00074800
C----- 00074900
COMMON / PICT / MIA, PIA, LSIA(32) 00075000
C 00075100
INTEGER PIA, LSIA 00075200
CHARACTER MIA=16 00075300
C 00075400
CHARACTER WK(5)=50, LOGM=50 00075500
C 00075600
C==> START 00075700
C 00075800
WK(1)='COPY '//SYSID//'.BAHAMA.PW '//1H'/' ' @'//PDATE(1:2) 00075900
+ //PDATE(4:5)//PDATE(7:8)'/'.WORK ' 00076000
WK(2)='ALLOC F(FT11F001) DA(@'//PDATE(1:2)//PDATE(4:5)'/ 00076100
+ PDATE(7:8)'/'.WORK) SHR ' 00076200
WK(3)='LIST '//1H'//SYSID//'.BAHAMA.DB '//1H'/' ' SYS(G) NON ' 00076300
WK(4)='LIST '//1H'//SYSID//'.BAHAMA.LOG '//1H'/' ' SYS(G) NON ' 00076400
WK(5)='LIST @'//PDATE(1:2)//PDATE(4:5)//PDATE(7:8)'/ 00076500
+ ' ' .WORK SYS(G) NON ' 00076600
C 00076700
LOGM='ALLOC F(FT30F001) '// 00076800
+ 'DA('//1H'//SYSID//'.BAHAMA.LOG '//1H'/'') SHR ' 00076900
C 00077000
MIA = 'SYSTEMU DB ' 00077100
IF ( ITYPE .NE. 1 ) MIA = 'SYSTEMU DB* ' 00077200
C 00077300
200 CONTINUE 00077400

```

DATA SET NAME : 'Q402#.AEG.FORT77(BAHAMA)'

```

.....1.....2.....3.....4.....5.....6.....7.....8
C DO 300 I=1,1251,50 00077500
300 BUFF(I:I+49)=50H 00077600
BUFF(1301:1312)=12H 00077700
C 00077800
CALL IPFHIO ( LSIA , MIA , BUFF , IRCA ) 00077900
C 00078000
IF ( IRCA .GT. 24 ) THEN 00078100
CALL ERROR ( 1019 ) 00078200
GO TO 999 00078300
END IF 00078400
C 00078500
IF ( IRCA .EQ. 3 ) GO TO 999 00078600
C 00078700
IF ( BUFF(1:1) .EQ. ' ' .AND. 00078800
+ BUFF(2:2) .EQ. ' ' .AND. 00078900
+ BUFF(3:3) .EQ. ' ' .AND. 00079000
+ BUFF(4:4) .EQ. ' ' ) THEN 00079100
CALL ERROR ( 1002 ) 00079200
GO TO 200 00079300
END IF 00079400
C 00079500
IF ( BUFF(1:1) .NE. ' ' ) CALL COMAND ( WK(3) ) 00079600
IF ( BUFF(2:2) .NE. ' ' ) CALL COMAND ( WK(4) ) 00079700
IF ( BUFF(3:3) .NE. ' ' ) THEN 00079800
CALL ULIST ( WK ) 00079900
CALL COMAND ( WK(5) ) 00080000
END IF 00080100
IF ( BUFF(4:4) .NE. ' ' ) THEN 00080200
CLOSE ( ILOG ) 00080300
CALL COMAND ( LOGM ) 00080400
OPEN ( ILOG ) 00080500
WRITE ( ILOG ) ' ' 00080600
CALL ERROR ( 1022 ) 00080700
END IF 00080800
GO TO 200 00080900
C 00081000
999 CONTINUE 00081100
RETURN 00081200
END 00081300
00081400

```

DATE : 88/10/25

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE DELDB * 00081500
C***** 00081600
SUBROUTINE DELDB ( IER ) 00081700
C 00081800
C=----- 00081900
C= FUNCTION = 00082000
C= TO DELETE THE DB RECORD AND ADDR RECORD = 00082100
C= PARAMETER = 00082200
C= IER : ERROR RETURN CODE (0) = 00082300
C=----- 00082400
C 00082500
C----- 00082600
C- PROCCSS NOW CONDITION - 00082700
C----- 00082800
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC 00082900
C 00083000
CHARACTER LOGID*8, PDATE*8, PASSWD*8 00083100
INTEGER*2 IUS*OW, IMISS, IFUNC 00083200
C----- 00083300
C- DATABASE MANAGEMENT AREA - 00083400
C----- 00083500
COMMON / MANAG / LRECD, LRMAX, LRNOW, ITMAX, ITNOW, 00083600
+ ITNAME(16), ITFORM(16), ITCHAR(16), 00083700
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID 00083800
C 00083900
CHARACTER ITNAME*24, ITFORM*24, SYSID*5 00084000
INTEGER*2 LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS, 00084100
+ ITEMPE, IDATAS, IDATAE 00084200
C----- 00084300
C- ADDR FILES DATA - 00084400
C----- 00084500
COMMON / ADDR* / NDBAD(32000), NITAD(16,32000), 00084600
+ NMATCH(32000),NCON, NDOM, 00084700
+ ISEL(32000), IPRO(16), INUM(16), 00084800
+ IS(16), IE(16), NKEY 00084900
C 00085000
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY 00085100
LOGICAL*1 ISEL, IPRO 00085200
C----- 00085300
C- UNIT INPUT OUTPUT NUMBER - 00085400
C----- 00085500
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST, 00085600
+ ITSSIN, ITSSOT, IWORK*,IPW 00085700
C 00085800
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT, 00085900
+ IWORK*,IPW 00086000
C----- 00086100
C- FILE INPUT/OUTPUT BUFFER - 00086200
C----- 00086300
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL 00086400
00086500
00086600
00086700
00086800
00086900

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C 00087000
CHARACTER BUFF*W*384, BUFF*1312, COND*384,CSEL*384 00087100
C----- 00087200
C- DB FILE EMPTY RECORD CONTROL - 00087300
C----- 00087400
COMMON / BITMAP / MAP 00087500
C 00087600
CHARACTER MAP*4000 00087700
C 00087800
IER = 0 00087900
C=----- 00088000
C- CONTROL BIT OFF & RECORD PUT & PRG AREA UPDATE 00088100
C----- 00088200
DO 200 I=1,LRNOW 00088300
IF ( ISEL(I) ) THEN 00088400
NBIT = NDBAD(I) - IDATAS + 1 00088500
CALL BONDFF ( MAP, NBIT, NBIT, 1, IB, 1, IERR ) 00088600
LRNOW = LRNOW - 1 00088700
CALL MAGPUT ( NDBAD(I) , IR ) 00088800
DO 100 J=1,LRNOW 00088900
NDBAD (J) = NDBAD (J+1) 00089000
NMATCH(J) = NMATCH(J+1) 00089100
ISEL (J) = ISEL (J+1) 00089200
DO 100 K=1,ITNOW 00089300
NITAD(K,J) = NITAD(K,J+1) 00089400
100 CONTINUE 00089500
CALL LOGOUT ( 1 , CSEL , IER ) 00089600
GO TO 999 00089700
END IF 00089800
200 CONTINUE 00089900
IER = 1 00090000
C 00090100
999 RETURN 00090200
END 00090300
00090400

```

183

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C SUBROUTINE DELETE *
C*****
SUBROUTINE DELETE
C
C-----
C- PROCESS NOW CONDITION
C-----
COMMON / PROSS / LOGID, PDATE, IUSWOW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUSWOW, IMISS, IFUNC
C-----
C- DATABASE MANAGEMENT AREA
C-----
COMMON / MANAG / LRECD, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE
C-----
C- ADDR FILES DATA
C-----
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C-----
C- PROCESS CONDITION AREA
C-----
COMMON / CONDIT / JC(3)
C
INTEGER*2 JC
C-----
C- FILE INPUT/OUTPUT BUFFER
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C-----
C- ERROR MESSAGE AREA
C-----
COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)
C
CHARACTER MFLAG*2, MSG*60

```

```

00090500
00090600
00090700
00090800
00090900
00091000
00091100
00091200
00091300
00091400
00091500
00091600
00091700
00091800
00091900
00092000
00092100
00092200
00092300
00092400
00092500
00092600
00092700
00092800
00092900
00093000
00093100
00093200
00093300
00093400
00093500
00093600
00093700
00093800
00093900
00094000
00094100
00094200
00094300
00094400
00094500
00094600
00094700
00094800
00094900
00095000
00095100
00095200
00095300
00095400
00095500
00095600
00095700
00095800
00095900

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

INTEGER*2 NMSG, NERCD
C-----
C- RELATIVE ADDRESS AREA
C-----
COMMON / ISOU / ISO
C
INTEGER*4 ISO
C-----
C- DELETE PROC USED DATA
C-----
CHARACTER PW*8, UID*8, UPW*8, OPW*8
INTEGER*4 IRAD(20)
C
5 CALL SEARCH
C
IDELNO = 0
IF( ISO .NE. 0 ) THEN
  ISTART = ISO
  IDIM = 1
  COND( IS(1):IE(12) ) = BUFF*W(19)(1:224)
  CALL GETDB( ISTART, IDIM, IEND, ICO, IRAD, IER )
  DO 10 J = 0, 360, 24
    COND( J+1:J+24 ) = 24H
10 CONTINUE
  COND( IS(1):IE(12) ) = BUFF*W(1)( IS(1):IE(12) )
  IF( LOGID(1:5) .NE. BUFF*W(1)(65:69) .AND.
  + LOGID(1:5) .NE. BUFF*W(1)(78:82) ) THEN
    WRITE(6,50) BUFF*W(1)(65:69), BUFF*W(1)(78:82)
50 FORMAT( ' PLEASE INPUT THE PASSWORD IN ',
  + A5, ' OR ', A5, ' ' )
  CALL PROMPT(2, ' * INPUT THE PASSWORD ==> ', IRC, 0)
  READ(5, '(A8)') PW
  Y = 0
  UID(1:5) = BUFF*W(1)(65:69)
  CALL BAHAPW( UID, UPW, OPW )
  IF( PW .NE. UPW .AND. PW .NE. OPW ) Y = Y + 1
  IF( BUFF*W(1)(78:82) .NE. 5H ) THEN
    UID(1:5) = BUFF*W(1)(78:82)
    CALL BAHAPW( UID, UPW, OPW )
    IF( PW .NE. UPW .AND. PW .NE. OPW ) Y = Y + 1
  END IF
ENDIF
IF( Y .LT. 2 ) THEN
  ISEL(ISO) = .TRUE.
  CALL DELDB( IER )
  IF( IER .EQ. 0 ) IDELNO = IDELNO + 1
  ISEL(ISO) = .FALSE.
  ISO = 0
ENDIF
ENDIF
IF( IDELNO .GT. 0 ) THEN
  WRITE(6,200) IDELNO
200 FORMAT( ' NUMBER OF DELETE RECORDS IS ', I4 )

```

```

00096000
00096100
00096200
00096300
00096400
00096500
00096600
00096700
00096800
00096900
00097000
00097100
00097200
00097300
00097400
00097500
00097600
00097700
00097800
00097900
00098000
00098100
00098200
00098300
00098400
00098500
00098600
00098700
00098800
00098900
00099000
00099100
00099200
00099300
00099400
00099500
00099600
00099700
00099800
00099900
00100000
00100100
00100200
00100300
00100400
00100500
00100600
00100700
00100800
00100900
00101000
00101100
00101200
00101300
00101400

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
      DO 300 I = 0 , 1200 , 100
          BUFF(I+1:I+100) = 100H
      +
300   CONTINUE
      BUFF(1301:1312) = 12H
      GOTO 5
      ENDIF
999   RETURN
      END

```

```

00101500
00101600
00101700
00101800
00101900
00102000
00102100
00102200
00102300

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*****
C*  SUBROUTINE ERROR ( IERROR )
C*****
      SUBROUTINE ERROR ( IERROR )
C
C-----
C=  FUNCTION
C=  TO PUT THE ERROR MESSAGE FROM IERROR
C=  PARAMETER
C=  IERROR ; ERROR CODE (I)
C-----
C
C-----
C-  ERROR MESSAGE AREA
C-----
      COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)
C
      CHARACTER MFLAG*2, MSG*60
      INTEGER*2 NMSG, NERCD
C-----
C-  UNIT INPUT OUTPUT NUMBER
C-----
      COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
      +             ITSSIN, ITSSOT, IWORK*, IPW
C
      INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
      +             IWORK*, IPW
C
      DO 100 I=1,NMSG
          IF ( IERROR .EQ. NERCD(I) ) THEN
              WRITE ( ITSSOT , 1000 ) MFLAG, MSG ( I )
              RETURN
          END IF
      100 CONTINUE
C
1000 FORMAT ( 1H ,A2,A60 )
      RETURN
      END

```

```

00102400
00102500
00102600
00102700
00102800
00102900
00103000
00103100
00103200
00103300
00103400
00103500
00103600
00103700
00103800
00103900
00104000
00104100
00104200
00104300
00104400
00104500
00104600
00104700
00104800
00104900
00105000
00105100
00105200
00105300
00105400
00105500
00105600
00105700
00105800
00105900
00106000
00106100
00106200

```

PNC N9520 88-022(2)

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE GETDB *
C*****
SUBROUTINE GETDB ( ISTART , IDIM , IEND , ICO , IRAD , IER )
C
C=====
C= FUNCTION
C= TO GET THE DB RECORD
C= PARAMETER
C= ISTART: GET START ADDR ADDRESS (RELATIVE) (I) =
C= IDIM : GETTING DIMENSION (I) =
C= -N : TO GET THE -DIMENSION =
C= +N : TO GET THE +DIMENSION =
C= IEND : GET END ADDR ADDRESS (RELATIVE) (O) =
C= IRAD : GETTED ADDR ADDRESS (RELATIVE) (O) =
C= ICO : GETTED NUMBER OF RECORD (O) =
C= IER : ERROR CODE (O) =
C= 0 : NORMAL END
C= 1 : END OF DATA ( MIN OR MAX ) =
C-----
C
INTEGER IRAD(20)
C-----
C- DATABASE MANAGEMENT AREA
C-----
COMMON / MANAG / LRECOB, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITENPE, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LRECOB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEHPE, IDATAS, IDATAE
C-----
C- ADDR FILES DATA
C-----
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOH,
+ ISEL(32000), IPRO(16), INUH(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOH, INUH, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C-----
C- UNIT INPUT OUTPUT NUMBER
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*, IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*, IPW
C-----

```

```

00106300
00106400
00106500
00106600
00106700
00106800
00106900
00107000
00107100
00107200
00107300
00107400
00107500
00107600
00107700
00107800
00107900
00108000
00108100
00108200
00108300
00108400
00108500
00108600
00108700
00108800
00108900
00109000
00109100
00109200
00109300
00109400
00109500
00109600
00109700
00109800
00109900
00110000
00110100
00110200
00110300
00110400
00110500
00110600
00110700
00110800
00110900
00111000
00111100
00111200
00111300
00111400
00111500
00111600
00111700

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C- FILE INPUT/OUTPUT BUFFER
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C
COMMON / MLEVEL / LEVEL(16)
C
LOGICAL*1 LEVEL
C
IER = 0
ICO = 0
IF ( ISTART .LT. 1 .OR. ISTART .GT. LRNOW ) THEN
IER = 1
GO TO 999
END IF
C
IF ( ABS(IDIM) .EQ. 0 ) THEN
IER = 2
GO TO 999
END IF
C
IMIN = LRNOW
LMIN = 18
LHAX = 2
LSPN = -1
LCO = 1
LSP = 1
IF ( IDIM .LE. 0 ) THEN
IMIN = 1
LMIN = 1
LNAX = 17
LSPN = 1
LCO = 18
LSP = -1
END IF
C= RECORD GET START
C= DO 300 IEND=ISTART,IMIN,IDIM
C= IF ( NMATCH(IEND) .NE. NCON+NKEY ) GO TO 300
C= READ ( IDB , '(A)', REC=NDBAD(IEND) ) BUFF*W(20)
J = 0
J2 = 0
DO 100 K=1,ITNOW
IF ( .NOT.LEVEL(K) .AND. IPRO(K) .AND.
COND(IS(K):IE(K)) .EQ. BUFF*W(20) (IS(K):IE(K)) ) THEN
J = J + 1
C
ELSE IF ( LEVEL(K) .AND. IPRO(K) ) THEN
DO 50 L=1,ITNOW
IF ( LEVEL(L) .AND.

```

```

00111800
00111900
00112000
00112100
00112200
00112300
00112400
00112500
00112600
00112700
00112800
00112900
00113000
00113100
00113200
00113300
00113400
00113500
00113600
00113700
00113800
00113900
00114000
00114100
00114200
00114300
00114400
00114500
00114600
00114700
00114800
00114900
00115000
00115100
00115200
00115300
00115400
00115500
00115600
00115700
00115800
00115900
00116000
00116100
00116200
00116300
00116400
00116500
00116600
00116700
00116800
00116900
00117000
00117100
00117200

```

DATA SET NAME : 'Q402#.AEG.FORT77(BAHAMA)'

DATE : 88/10/25

.....1.....2.....3.....4.....5.....6.....7.....8

```

C****+      L.NE.K          .AND.          00117300
+          BUFF*W(20)(IS(L):IE(L)) .NE.    00117400
+          20H              .AND.          00117500
+          COND(IS(K):IE(K)).EQ.BUFF*W(20)(IS(L):IE(L)) ) 00117600
+          J2= J2+ 1        00117700
50          CONTINUE      00117800
          END IF          00117900
C          100 CONTINUE  00118000
C          IF ( J .EQ. NCON .AND.
+            ( NKEY .NE. 0 .AND. J2 .GE. 1 .OR.
+            NKEY .EQ. 0 ) ) THEN
          LCO = LCO + LSP  00118100
          DO 200 M=LHIN,LCO,LSPN 00118200
          IRAD(M) = IRAD(M+LSPN) 00118300
200        BUFF*W(M) = BUFF*W(M+LSPN) 00118400
          BUFF*W(LCO+LSPN) = BUFF*W(20) 00118500
          ICO = ICO + 1    00118600
          IRAD(LCO+LSPN) = IEND 00118700
          IF ( ICO .GE. 18 ) GO TO 999 00118800
          ELSE
          NMATCH(IEND) = 0 00118900
          NDOM = NDOM - 1  00119000
          END IF          00119100
C          300 CONTINUE  00119200
          IER = 3         00119300
C          999 CONTINUE  00119400
C=> IEND = IEND + LSPN  00119500
C          RETURN      00119600
          END           00119700
                    00119800
                    00119900
                    00120000
                    00120100
                    00120200
                    00120300
                    00120400
                    00120500
                    00120600

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE GETMSG *
C*****
SUBROUTINE GETMSG
C
C=====
C= FUNCTION =
C= TO READ THE MESSAGE FILE =
C=====
C
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*,IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*,IPW
C-----
C- ERROR MESSAGE AREA -
C-----
COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)
C
CHARACTER MFLAG*2, MSG*60
INTEGER*2 NMSG, NERCD
C
NMSG = 0
C
100 CONTINUE
NMSG = NMSG + 1
IF ( NMSG .GE. 51 ) GO TO 9999
READ ( IMSG , 10000 , END=9999 ) NERCD(NMSG) , MSG(NMSG)
GO TO 100
C
9999 CONTINUE
NMSG = NMSG - 1
RETURN
10000 FORMAT( I4 , T1 , A60 )
END

```

```

00120700
00120800
00120900
00121000
00121100
00121200
00121300
00121400
00121500
00121600
00121700
00121800
00121900
00122000
00122100
00122200
00122300
00122400
00122500
00122600
00122700
00122800
00122900
00123000
00123100
00123200
00123300
00123400
00123500
00123600
00123700
00123800
00123900
00124000
00124100
00124200
00124300
00124400
00124500
00124600

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE HEDER *
C*****
SUBROUTINE HEDER ( IN )
C
C=====
C= FUNCTION =
C= TO LISTUP THE HEDER LABEL =
C= PARAMETER =
C= IN ; USER OR OWNER FLAG =
C= 1 : USER =
C= 2 : OWNER =
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*,IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*,IPW
C-----
C- FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*(20), BUFF, COND, CSEL
C
CHARACTER BUFF*(384), BUFF=1312, COND=384, CSEL=384
C
INTEGER*2 IN
NCHARACTER USER(2)*6
NCHARACTER HED=44
CHARACTER VL=11
C
DATA VL / 'V-00 / L-00' /
DATA USER / NC' 利用者', NC' 管理者' /
DATA HED( 1:12) / NC' テクニカル リソース' /
DATA HED(13:24) / NC' 管理 システム' /
DATA HED(25:40) / NC' ( リスト )' /
C
HED(27:32) = USER(IN)(1:6)
WRITE ( ILIST , 100 ) HED , VL
C
100 FORMAT ( 1H1 ,/,1H ,2X,
+ '*** B A H A M A ***' , 5X ,
+ N40.H1 , 2X , A11 )
C
RETURN
END

```

```

00124700
00124800
00124900
00125000
00125100
00125200
00125300
00125400
00125500
00125600
00125700
00125800
00125900
00126000
00126100
00126200
00126300
00126400
00126500
00126600
00126700
00126800
00126900
00127000
00127100
00127200
00127300
00127400
00127500
00127600
00127700
00127800
00127900
00128000
00128100
00128200
00128300
00128400
00128500
00128600
00128700
00128800
00128900
00129000
00129100
00129200
00129300
00129400
00129500
00129600

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE LABEL *
C*****
SUBROUTINE LABEL ( IN )
C
C=====
C= FUNCTION =
C= TO LISTUP THE DATA ITEM'S LABEL =
C= PARAMETER =
C= IN ; USER OR OWNER FLAG =
C= 1 : USER =
C= 2 : OWNER =
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*,IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*,IPW
C-----
C- FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*(20), BUFF, COND, CSEL
C
CHARACTER BUFF*(384), BUFF=1312, COND=384, CSEL=384
C
NCHARACTER DS=8 , SYU=6 , TOU=3 , KOU=3 , SAK=3 , KEN=4 , BIK=3,
+ HIZ=3
C
DATA DS / NC'ファイル名' /
DATA SYU/ NC'ファイル種類' /
DATA TOU/ NC'登録者' /
DATA KOU/ NC'更新者' /
DATA SAK/ NC'削除者' /
DATA HIZ/ NC'日付' /
DATA KEN/ NC'検索項目' /
DATA BIK/ NC'備考' /
C=
C= USER ITEM LABEL
C=
C IF ( IN .EQ. 1 ) THEN
WRITE ( ILIST , 1000 ) DS , SYU , TOU , KOU , KEN , BIK
WRITE ( ILIST , 1100 ) HIZ , HIZ
GO TO 999
END IF
C=
C= OWNER ITEM LABEL
C=
C IF ( IN .EQ. 2 ) THEN
WRITE ( ILIST , 2000 ) DS , SYU , TOU , KOU , SAK , KEN , BIK00135000
WRITE ( ILIST , 2100 ) HIZ , HIZ , HIZ

```

```

00129700
00129800
00129900
00130000
00130100
00130200
00130300
00130400
00130500
00130600
00130700
00130800
00130900
00131000
00131100
00131200
00131300
00131400
00131500
00131600
00131700
00131800
00131900
00132000
00132100
00132200
00132300
00132400
00132500
00132600
00132700
00132800
00132900
00133000
00133100
00133200
00133300
00133400
00133500
00133600
00133700
00133800
00133900
00134000
00134100
00134200
00134300
00134400
00134500
00134600
00134700
00134800
00134900
00135000
00135100

```

∞

```

.....1.....2.....3.....4.....5.....6.....7.....8
C          END IF                                00135200
C          999 CONTINUE                          00135300
C          1000 FORMAT ( // , 1H , 12X , N8.M1 , 19X , N6.M1 , 8X , N3.M1 ,
+          3X , N3.M1 , 4X , N4.M1 , 13X , N3.M1 ) 00135400
C          1100 FORMAT ( 1H , 67X , 2(N3.M1,3X) ) 00135500
C          C2000 FORMAT ( // , 1H , 5X , N7.M1 , 27X , N2.M1 , 3(3X,N3.M1) ,
+          8X , N6.M1 , 9X , N3.M1 )              00135600
C          C2100 FORMAT ( 1H , 53X , 3(N3.M1,3X) ) 00135700
C          RETURN                                00135800
C          END                                  00135900

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*****
C* SUBROUTINE LISTUP * 00136600
C***** 00136700
SUBROUTINE LISTUP ( IRC ) 00136800
C 00136900
C= FUNCTION = 00137000
C= TO LISTUP THE SEARCH DATAS = 00137100
C= PARAMETER = 00137200
C= IRC : ERROR CODE (0) = 00137300
C----- 00137400
C- PROCESS NOW CONDITION - 00137500
C----- 00137600
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC 00137700
C 00137800
CHARACTER LOGID*8, PDATE*8, PASSWD*8 00137900
INTEGER*2 IUS*OW, IMISS, IFUNC 00138000
C----- 00138100
C- DATABASE MANAGEMENT AREA - 00138200
C----- 00138300
COMMON / MANAG / LRECDB, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID 00138400
C 00138500
CHARACTER ITNAME*24, ITFORM*24, SYSID*5 00138600
INTEGER*2 LRECDB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE 00138700
C----- 00138800
C- UNIT INPUT OUTPUT NUMBER - 00138900
C----- 00139000
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*,IPW 00139100
C 00139200
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*,IPW 00139300
C----- 00139400
C- ADDR FILES DATA - 00139500
C----- 00139600
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000),NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY 00139700
C 00139800
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY 00139900
LOGICAL*1 ISEL, IPRO 00140000
C----- 00140100
C- FILE INPUT/OUTPUT BUFFER - 00140200
C----- 00140300
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL 00140400

```


.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE LOGOUT * 00150600
C***** 00150700
SUBROUTINE LOGOUT ( IFL , AREA , IER ) 00150800
00150900
C 00151000
C----- 00151100
C= FUNCTION = 00151200
C= TO PUT THE LOG FILE = 00151300
C= PARAMETER = 00151400
C= IFL : PROCESS FLAG (I) = 00151500
C= 1 : DELETE = 00151600
C= 2 : UPDATE = 00151700
C= AREA: PUT AREA (I) = 00151800
C= IER : ERROR RETURN CODE (O) = 00151900
C----- 00152000
C 00152100
C----- 00152200
C- PROCCSS NOW CONDITION - 00152300
C----- 00152400
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC 00152500
C 00152600
CHARACTER LOGID*8, PDATE*8, PASSWD*8 00152700
INTEGER*2 IUS*OW, IMISS, IFUNC 00152800
C 00152900
C----- 00153000
C- UNIT INPUT OUTPUT NUMBER - 00153100
C----- 00153200
COMMON / UNIT / IDB, IADDR, ILOG, INSG, IWORK, ILIST, 00153300
+ ITSSIN, ITSSOT, IWORK*,IPW 00153400
C 00153500
INTEGER*2 IDB, IADDR, ILOG, INSG, IWORK, ILIST, ITSSIN, ITSSOT, 00153600
+ IWORK*,IPW 00153700
C 00153800
CHARACTER AREA*(*) 00153900
C 00154000
IER = 0 00154100
C= 00154200
C= TO PUT THE LOG FILE 00154300
C= 00154400
WRITE( ILOG, * ) IFL , LOGID , PDATE , AREA 00154500
C 00154600
999 RETURN 00154700
END 00154800

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C***** 00154900
C* SUBROUTINE MAGPUT * 00155000
C***** 00155100
SUBROUTINE MAGPUT( IAD , IER ) 00155200
C 00155300
C----- 00155400
C= FUNCTION = 00155500
C= TO PUT THE SPACE MANAGEMENT RECORD = 00155600
C= PARAMETER = 00155700
C= IAD : RECORD ADDRESS (I) = 00155800
C= IER : ERROR RETURN CODE (O) = 00155900
C----- 00156000
C 00156100
INTEGER*2 IAD 00156200
C 00156300
C----- 00156400
C- PROCESS NOW CONDITION - 00156500
C----- 00156600
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC 00156700
C 00156800
CHARACTER LOGID*8, PDATE*8, PASSWD*8 00156900
INTEGER*2 IUS*OW, IMISS, IFUNC 00157000
C 00157100
C----- 00157200
C- DATABASE MANAGEMENT AREA - 00157300
C----- 00157400
COMMON / MANAG / LRECCB, LRMAX, LRNOW, ITMAX, ITNOW, 00157500
+ ITNAME(16), ITFORM(16), ITCHAR(16), 00157600
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID 00157700
C 00157800
CHARACTER ITNAME*24, ITFORM*24, SYSID*5 00157900
INTEGER*2 LRECCB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS, 00158000
+ ITEMPE, IDATAS, IDATAE 00158100
C 00158200
C----- 00158300
C- ADDR FILES DATA - 00158400
C----- 00158500
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000), 00158600
+ NMATCH(32000),NCON, NDOM, 00158700
+ ISEL(32000), IPRO(16), INUM(16), 00158800
+ IS(16), IE(16), NKEY 00158900
C 00159000
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY 00159100
LOGICAL*1 ISEL, IPRO 00159200
C 00159300
C----- 00159400
C- UNIT INPUT OUTPUT NUMBER - 00159500
C----- 00159600
COMMON / UNIT / IDB, IADDR, ILOG, INSG, IWORK, ILIST, 00159700
+ ITSSIN, ITSSOT, IWORK*,IPW 00159800
C 00159900
INTEGER*2 IDB, IADDR, ILOG, INSG, IWORK, ILIST, ITSSIN, ITSSOT, 00160000
+ IWORK*,IPW 00160100
C 00160200
C----- 00160300

```

-01-

```

.....1.....2.....3.....4.....5.....6.....7.....8
C CHARACTER BUFF*W=384, BUFF*1312, COND*384, CSEL=384 00142100
C COMMON / MLEVEL / LEVEL(16) 00142200
C LOGICAL*1 LEVEL 00142300
C IF ( IFUNC .LT. 0 .OR. IFUNC .GT. 2 ) THEN 00142400
  IRC = 1 00142500
  GO TO 999 00142600
  END IF 00142700
C IPTOT = (NDOH-1) * 0.125 + 1 00142800
C INP = 1 00142900
C DO 1000 I=1,LRNOW 00143000
  IF ( NMATCH(I) .NE. NCON+NKEY ) GO TO 1000 00143100
  READ ( IOB , '(A)', REC=NDBAD(I) ) BUFF*W(20) 00143200
  J = 0 00143300
  J2= 0 00143400
  DO 100 K=1,ITHOW 00143500
    IF ( .NOT.LEVEL(K).AND.IPRO(K) .AND. 00143600
      + COND(IS(K):IE(K)) .EQ. BUFF*W(20)(IS(K):IE(K)) ) THEN 00143700
      J = J + 1 00143800
    ELSE IF ( LEVEL(K).AND.IPRO(K) ) THEN 00143900
      DO 50 L=1,ITHOW 00144000
        IF ( LEVEL(L) .AND. 00144100
          + L.NE.K .AND. 00144200
          + BUFF*W(20)(IS(L):IE(L)) .NE. 00144300
          + 20H .AND. 00144400
          + COND(IS(K):IE(K)) .EQ. BUFF*W(20)(IS(L):IE(L)) ) 00144500
          J2= J2+ 1 00144600
        50 CONTINUE 00144700
      END IF 00144800
    C****+ 00144900
    + 00145000
    + 00145100
    + 00145200
    + 00145300
  50 CONTINUE 00145400
  END IF 00145500
C 100 CONTINUE 00145600
C IF ( J .EQ. NCON .AND. 00145700
  + ( NKEY .NE. 0 .AND. J2 .GE. 1 .OR. 00145800
  + NKEY .EQ. 0 ) ) THEN 00145900
C JP = MOD ( INP , 8 ) 00146000
  IF ( JP .EQ. 1 ) THEN 00146100
    INPAGE = (INP-1) * 0.125 + 1 00146200
    CALL HEDER ( IFUNC ) 00146300
    CALL PAGE ( IPTOT , INPAGE , PDATE ) 00146400
    CALL LABEL ( IFUNC ) 00146500
  END IF 00146600
  INP = INP + 1 00146700
  WRITE ( ILIST , 10000 ) 00146800
  + BUFF*W(20)(IS(1):IE(1)), BUFF*W(20)(IS(2):IE(2)), 00146900
  + BUFF*W(20)(IS(3):IE(3)), BUFF*W(20)(IS(5):IE(5)), 00147000
  + BUFF*W(20)(IS(7):IE(7)), BUFF*W(20)(IS(12):IE(12))-25) 00147100
  WRITE ( ILIST , 11000 ) 00147200
  00147300
  00147400
  00147500

```

DATA SET NAME : 'Q402#.AEG.FORT77(BAHAMA)'

```

.....1.....2.....3.....4.....5.....6.....7.....8
+ BUFF*W(20)(IS(4):IE(4)), BUFF*W(20)(IS(6):IE(6)), 00147600
+ BUFF*W(20)(IS(8):IE(8)), BUFF*W(20)(IS(12)+25:IE(12)) 00147700
WRITE ( ILIST , 12000 ) 00147800
+ (BUFF*W(20)(IS(K):IE(K)),K=9,11) 00147900
ELSE 00148000
* NMATCH(I) = 0 00148100
  NDOH = NDOH - 1 00148200
END IF 00148300
C 1000 CONTINUE 00148400
C 9999 CONTINUE 00148500
C= LISTUP THE TOTAL NO. OF DATA SETS 00148600
C= WRITE ( ILIST , 30000 ) NDOH 00148700
C 10000 FORMAT (/1H , 1X, A44, 1X, A20, 1X, A8, 1X, A8, 1X, A20, 1X, A25) 00148800
11000 FORMAT ( 1H , 67X, A8, 1X, A8, 1X, A20, 1X, A25) 00148900
12000 FORMAT ( 2(1H ,85X, A20 /) , 1H , 85X , A20 ) 00149000
20000 FORMAT (/1H ,1X,A44,1X,A6,1X,A5,4X,A5,16X,A20,2X,A25) 00149100
20100 FORMAT (/1H ,1X,A44,1X,A6,1X,A5,13X,A5,7X,A20,2X,A25) 00149200
21000 FORMAT ( 1H ,53X,A8,1X,A8,13X,A20,2X,A25) 00149300
21100 FORMAT ( 1H ,53X,A8,10X,A8,4X,A20,2X,A25) 00149400
22000 FORMAT ( 2(1H ,83X, A20 /) , 1H ,83X , A20 ) 00149500
30000 FORMAT ( //,1H , ' *** TOTAL NUMBER OF FILES ****>> ' , 15 , 00149600
  + ' ***' ) 00149700
C 999 RETURN 00150000
END 00150100
00150200
00150300
00150400
00150500

```

DATE : 88/10/25

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE LOGOUT *
C*****
SUBROUTINE LOGOUT ( IFL , AREA , IER )
C
C-----
C= FUNCTION =
C= TO PUT THE LOG FILE =
C= PARAMETER =
C= IFL : PROCESS FLAG (I) =
C= 1 : DELETE =
C= 2 : UPDATE =
C= AREA: PUT AREA (I) =
C= IER : ERROR RETURN CODE (O) =
C-----
C- PROCCES NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUS*OW, IMISS, IFUNC
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*,IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*,IPW
C
CHARACTER AREA*(*)
C
IER = 0
C=
C= TO PUT THE LOG FILE
C=
WRITE( ILOG, * ) IFL , LOGID , PDATE , AREA
C
999 RETURN
END

```

```

00150600
00150700
00150800
00150900
00151000
00151100
00151200
00151300
00151400
00151500
00151600
00151700
00151800
00151900
00152000
00152100
00152200
00152300
00152400
00152500
00152600
00152700
00152800
00152900
00153000
00153100
00153200
00153300
00153400
00153500
00153600
00153700
00153800
00153900
00154000
00154100
00154200
00154300
00154400
00154500
00154600
00154700
00154800

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C= SUBROUTINE MAGPUT *
C*****
SUBROUTINE MAGPUT( IAD , IER )
C
C-----
C= FUNCTION =
C= TO PUT THE SPACE MANAGEMENT RECORD =
C= PARAMETER =
C= IAD : RECORD ADDRESS (I) =
C= IER : ERROR RETURN CODE (O) =
C-----
C
INTEGER*2 IAD
C-----
C- PROCCES NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUS*OW, IMISS, IFUNC
C-----
C- DATABASE MANAGEMENT AREA -
C-----
COMMON / MANAG / LRECD, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE
C-----
C- ADDR FILES DATA -
C-----
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000),NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*,IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*,IPW
C-----

```

```

00154900
00155000
00155100
00155200
00155300
00155400
00155500
00155600
00155700
00155800
00155900
00156000
00156100
00156200
00156300
00156400
00156500
00156600
00156700
00156800
00156900
00157000
00157100
00157200
00157300
00157400
00157500
00157600
00157700
00157800
00157900
00158000
00158100
00158200
00158300
00158400
00158500
00158600
00158700
00158800
00158900
00159000
00159100
00159200
00159300
00159400
00159500
00159600
00159700
00159800
00159900
00160000
00160100
00160200
00160300

```



```

.....1.....2.....3.....4.....5.....6.....7.....8
C- FILE INPUT/OUTPUT BUFFER - 00160400
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL 00160600
C 00160700
CHARACTER BUFF*W=384, BUFF=1312, COND=384, CSEL=384 00160800
C 00160900
C-----
C- DB FILE EMPTY RECORD CONTROL - 00161000
C-----
COMMON / BITMAP / MAP 00161100
C 00161200
CHARACTER MAP=4000 00161300
C 00161400
IF ( IAD .LE. 0 .OR. IAD .GT. LRMAX+IDATAS-1 ) THEN 00161500
IER = 1 00161600
GO TO 999 00161700
END IF 00161800
C 00161900
IBIT = IAD - IDATAS + 1 00162000
LR = (IBIT-1) / LRECD + 1 00162100
LS = (LR-1) * LRECD + 1 00162200
LR = LR + ITEMPS - 1 00162300
C= 00162400
C= BIT MAP RECORD PUT 00162500
C= 00162600
WRITE( IDB , '(3A128)' , REC=LR ) 00162700
+ MAP(LS:LS+127), 00162800
+ MAP(LS+128:LS+255), 00162900
+ MAP(LS+256:LS+383) 00163000
C= 00163100
CONTROL RECORD PUT 00163200
C= 00163300
WRITE( IDB , '(3I5,2I2,3I5)' , REC=1 ) 00163400
+ LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITEMPS, ITEMPE, IDATAS 00163500
C 00163600
999 RETURN 00163700
END 00163800
00163900
00164000

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C ***** 00164100
C * BA HA MA * 00164200
C * <<BASE DATA HANDRING MANAGEMENT SYSTEM>> * 00164300
C * * 00164400
C * THIS PROGRAM IS MAIN OF BASE DATA HANDRING MANAGEMENT SYSTEM. * 00164500
C * * 00164600
C * FUNCTION * 00164700
C * 1 : USER SUBSYSTEM * 00164800
C * (1). TO ADD THE USER INFORMATION TO DB. * 00164900
C * (2). TO UPDATE THE USER INFORMATION TO DB. * 00165000
C * (3). TO SEARCH THE USER INFORMATION FROM DB. * 00165100
C * (4). TO DELETE THE USER INFORMAION OF DB. * 00165200
C * 2 : OWNER SUBSYSTEM * 00165300
C * (1). TO ADD THE USER TO PASSWORD FILE. * 00165400
C * (2). TO UPDATE THE USER TO PASSWORD FILE. * 00165500
C * (3). TO SEARCH THE USER FROM PASSWORD FILE. * 00165600
C * (4). TO DELETE THE USER OF PASSWORD FILE. * 00165700
C * (5). TO LIST UP THE DB FILE. * 00165800
C * (6). TO LIST UP THE LOG FILE. * 00165900
C * (7). TO LIST UP THE USER PASSWORD FILE. * 00166000
C * (8). TO INITIALIZE TO LOG FILE. * 00166100
C * * 00166200
C * DATE * 00166300
C * 07/1988 : BAHAMA VOO/LOO * 00166400
C * * 00166500
C * WORKING GROUP * 00166600
C * WATANABE : MAIN , MANAGER EXECUTE & FILE I/O ROUTINE. * 00166700
C * SUKEGAWA : * 00166800
C * SUZUKI : ADD,UPDATE,SEARCH & DELETE CONTROL ROUTINE. * 00166900
C * * 00167000
C ***** 00167100
C-----
C- PROCCES NOW CONDIITION - 00167200
C-----
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC 00167300
C 00167400
CHARACTER LOGID*8, PDATE*8, PASSWD*8 00167500
INTEGER*2 IUS*OW, IMISS, IFUNC 00167600
C 00167700
C-----
C- UNIT INPUT OUTPUT NUMBER - 00167800
C-----
COMMON / UNIT / IDB, IADDR, ILOG, MSG, IWORK, ILIST, 00167900
+ ITSSIN, ITSSOT, IWORK*, IPW 00168000
C 00168100
INTEGER*2 IDB, IADDR, ILOG, MSG, IWORK, ILIST, ITSSIN, ITSSOT, 00168200
+ IWORK*,IPW 00168300
C 00168400
C-----
C- PICTURE COMMON AREA - 00168500
C-----
COMMON / PICT / MIA , PIA , LSIA(32) 00168600
C 00168700
CHARACTER MIA*16 00168800
00168900
00169000
00169100
00169200
00169300
00169400
00169500

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C          00169600
C----- 00169700
C-  TERMTYPE COMMON AREA  00169800
C----- 00169900
COMMON / TERM / IRO , IR1 , TNAME , ITYPE , IRCD 00170000
C          00170100
C CHARACTER TNAME*8 00170200
C          00170300
CHARACTER TPW*8, DUMMY*8
C          00170400
CHARACTER CHD1*80 00170500
C          00170600
C=== 00170700
ERROR MESSAGE SET 00170800
C=== 00170900
CALL GETMSG 00171000
C=== 00171100
TO SET THE TODAY'S DATE 00171200
C=== 00171300
CALL DATE ( PDATE ) 00171400
C          00171500
C=== 00171600
LOGON ID & PASSWORD CHECK 00171700
C=== 00171800
CALL INPRO ( '* TYPE IN YOURE PASSWORD ==>' , 1 ) 00171900
100 READ ( ITSSIN , '(A8)' , END=99999 ) TPW 00172000
C          00172100
PASSWD = 8H 00172200
C          00172300
CALL GETJN ( LOGID , DUMMY ) 00172400
C          00172500
CALL P*W* ( TPW , IRC ) 00172600
IF ( IRC .EQ. 1 ) GO TO 100 00172700
C          00172800
IF ( PASSWD .EQ. 8H ) THEN 00172900
CALL ERROR ( 1005 ) 00173000
GO TO 99999 00173100
END IF 00173200
C=== 00173300
TO GET THE TERMINAL TYPE 00173400
C=== 00173500
CALL TRMTYP ( IRO , IR1 , TNAME , ITYPE , IRCD ) 00173600
C          00173700
IF ( ITYPE .NE. 1 .AND. ITYPE .NE. 2 ) THEN 00173800
CALL ERROR ( 1010 ) 00173900
GO TO 99999 00174000
END IF 00174100
C=== 00174200
TO ALLOCATE THE IPF MENULIB 00174300
C=== 00174400
CMD1 = 'ALLOC DA(''Q401#.IPF.DATA'' ) F(SYSMENU) SHR REU ' 00174500
CALL IPFCHD ( IRC1, IRC2, CHD1, 50 ) 00174600
C          00174700
PIA = 0 00174800
DO 200 I=1,32 00174900
LSIA(I) = 0 00175000

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

200 CONTINUE 00175100
C=== 00175200
TO OPEN THE IPF LIBRALY 00175300
C=== 00175400
CALL IPFOVS ( PIA, LSIA, IRCA ) 00175500
IF( IRCA .NE. 0 ) THEN 00175600
CALL ERROR ( 1003 ) 00175700
GO TO 99999 00175800
ENDIF 00175900
CALL IPFPFF ( LSIA , IRCA ) 00176000
C=== 00176100
TO OPEN THE DB 00176200
C=== 00176300
CALL DBACCE ( 1 , IRO ) 00176400
C          00176500
IFUNC = 0 00176600
C          00176700
END USER OPEN 00176800
C          00176900
IF ( IUS*OW .EQ. 1 ) THEN 00177000
IFUNC = 1 00177100
CALL SELECT 00177200
GO TO 88888 00177300
END IF 00177400
C          00177500
OWNER USER OPEN 00177600
C          00177700
IF ( IUS*OW .EQ. 2 ) THEN 00177800
IFUNC = 2 00177900
CALL MENU 00178000
GO TO 88888 00178100
END IF 00178200
C          00178300
C===>DEBUG 00178400
* WRITE(6,*) IFUNC 00178500
C===>END 00178600
88888 CONTINUE 00178700
CALL ERROR ( 1001 ) 00178800
C          00178900
DATA BASE CLOSE 00179000
C          00179100
CALL DBACCE ( 2 , IRC ) 00179200
C          00179300
IPF PICTURE CLOSE 00179400
C          00179500
CALL IPFCVS (LSIA , IRCA) 00179600
C          00179700
99999 CONTINUE 00179800
STOP ' 00179900
END GOOD BY SEE YOU AGEIN. ' 00180000
00180100
00180200

```

- 03 -

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*****
C= SUBROUTINE MATCHC *
C*****
SUBROUTINE MATCHC
C
C=====
C= FUNCTION =
C= TO SET THE MATCH LEVEL IN /ADDRVR/ =
C=====
C-----
C- DATABASE MANAGEMENT AREA -
C-----
COMMON / MANAG / LRECOB, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMP, ITEHPE, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LRECOB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMP,
+ ITEMP, IDATAS, IDATAE
C
C-----
C- ADDR FILES DATA -
C-----
COMMON / ADDRVR / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C-----
C- MATCH CHECK LEVEL KEY -
C-----
COMMON / MLEVEL / LEVEL(16)
C
LOGICAL*1 LEVEL
C
C=
C= TO CHECK THE MATCH LEVEL
C=
NDOM = 0
DO 800 I=1,LRNOW
NMATCH(I) = 0
NDUMMY = 0
DO 500 J=1,ITNOW
IF (.NOT.LEVEL(J).AND.IPRO(J).AND.INUM(J).EQ.NITAD(J,I)) THEN
NMATCH(I) = NMATCH(I) + 1
C
ELSE IF
+ ( LEVEL(J).AND.IPRO(J) ) THEN
DO 400 K=1,ITNOW
IF ( LEVEL(K) .AND.
C****+ K.NE.J .AND.
+ NITAD(K,I).NE.0 .AND.
00180300
00180400
00180500
00180600
00180700
00180800
00180900
00181000
00181100
00181200
00181300
00181400
00181500
00181600
00181700
00181800
00181900
00182000
00182100
00182200
00182300
00182400
00182500
00182600
00182700
00182800
00182900
00183000
00183100
00183200
00183300
00183400
00183500
00183600
00183700
00183800
00183900
00184000
00184100
00184200
00184300
00184400
00184500
00184600
00184700
00184800
00184900
00185000
00185100
00185200
00185300
00185400
00185500
00185600
00185700

```

DATA SET NAME : 'Q402#.AEG.FORT77<BAHAMA>'

```

.....1.....2.....3.....4.....5.....6.....7.....8
+ INUM(J).EQ.NITAD(K,I) ) THEN
NDUMMY = NDUMMY + 1
CDS
* WRITE(6,*) ' MATCHC: LEVEL K J INUM NITAD ',
* + LEVEL(K),K,J,INUM(J),NITAD(K,I)
CDE
END IF
400 CONTINUE
END IF
500 CONTINUE
IF ( NMATCH(I) .EQ. NCON .AND.
+ ( NKEY.NE.0.AND.NDUMMY.GE.1 .OR.
+ NKEY.EQ.0 ) ) THEN
NMATCH(I) = NCON + NKEY
NDOM = NDOM + 1
ELSE
NMATCH(I) = 0
END IF
C
800 CONTINUE
C
RETURN
END
00185800
00185900
00186000
00186100
00186200
00186300
00186400
00186500
00186600
00186700
00186800
00186900
00187000
00187100
00187200
00187300
00187400
00187500
00187600
00187700
00187800
00187900
00188000

```

DATE : 88/10/25

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE MATCHS *
C*****
SUBROUTINE MATCHS
C
C-----
C= FUNCTION =
C= TO CHECK THE ADDR INFOMATION & SET THE MATCH=
C= PARAMETER =
C= NUM ; NUMBER OF DATA MATCH =
C-----
C
C-----
C- PROCCSS NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUSVOW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUSVOW, IMISS, IFUNC
C
C-----
C- DATABASE MANAGEMENT AREA -
C-----
COMMON / MANAG / LREADB, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LREADB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE
C
C-----
C- ADDR FILES DATA -
C-----
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000),NCON, NDOH,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOH, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C
C-----
C- FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF=1312, COND*384, CSEL*384
C
C-----
C- MATCH LEVEL CHECK -
C-----
COMMON / MLEVEL / LEVEL(16)
C
LOGICAL*1 LEVEL

```

```

00188100
00188200
00188300
00188400
00188500
00188600
00188700
00188800
00188900
00189000
00189100
00189200
00189300
00189400
00189500
00189600
00189700
00189800
00189900
00190000
00190100
00190200
00190300
00190400
00190500
00190600
00190700
00190800
00190900
00191000
00191100
00191200
00191300
00191400
00191500
00191600
00191700
00191800
00191900
00192000
00192100
00192200
00192300
00192400
00192500
00192600
00192700
00192800
00192900
00193000
00193100
00193200
00193300
00193400
00193500

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C CHARACTER*50 BLANK
DATA BLANK /'
C
NCON = 0
NKEY = 0
I1 = 0
I2 = 0
C
DO 200 I=1,ITNOW
IPRO(I) = .FALSE.
I1 = I2 + 1
I2 = I1 + ITCHAR(I) - 1
INUM(I) = 0
IF ( COND(I1:I2) .NE. BLANK(1:I2-I1+1) ) THEN
IPRO(I) = .TRUE.
CALL CHANUM ( COND(I1:I2) , INUM(I) )
IF ( LEVEL(I) ) THEN
NKEY = NKEY + 1
ELSE
NCON = NCON + 1
END IF
CDS
* WRITE(6,*) ' MATCHS COND,INUM,NCON',COND(I1:I2),INUM(I),NCON
CODE
END IF
200 CONTINUE
C
RETURN
END

```

```

00193600
00193700
00193800
00193900
00194000
00194100
00194200
00194300
00194400
00194500
00194600
00194700
00194800
00194900
00195000
00195100
00195200
00195300
00195400
00195500
00195600
00195700
00195800
00195900
00196000
00196100
00196200
00196300
00196400
00196500

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C*  SUBROUTINE MENU *
C*****
SUBROUTINE MENU
C
C-----
C  FUNCTION =
C  TO DISPLAY THE INITIAL PICTURE. =
C-----
C
C-----
C-  PROCCSS NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUSWOW, PASSWD, IMISS, IFUNC
C
C  CHARACTER LOGID=8, PDATE=8, PASSWD=8
C  INTEGER=2 IUSWOW, IMISS, IFUNC
C-----
C-  FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
C  CHARACTER BUFF*W=384, BUFF=1312, COND=384, CSEL=384
C-----
C-  TERMTYPE COMMON AREA -
C-----
COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD
C
C  CHARACTER TNAME=8
C-----
C-  PICTURE COMMON AREA -
C-----
COMMON / PICT / MIA, PIA, LSIA(32)
C
C  CHARACTER MIA=16
C
200 CONTINUE
C===
C=== TO DISPLAY THE INITIAL MENU
C===
MIA = 'SYSTEMU MENU '
IF ( ITYPE .NE. 1 ) MIA = 'SYSTEMU MENUW '
C
250 CONTINUE
DD 300 I=1,1251,50
300 BUFF(I:I+49)=50H
BUFF(1301:1312)=12H
IFUNC = 999
C
CALL IPFMIO ( LSIA, MIA, BUFF, IRCA )
C
IF( IRCA .GT. 24 ) THEN

```

```

00196600
00196700
00196800
00196900
00197000
00197100
00197200
00197300
00197400
00197500
00197600
00197700
00197800
00197900
00198000
00198100
00198200
00198300
00198400
00198500
00198600
00198700
00198800
00198900
00199000
00199100
00199200
00199300
00199400
00199500
00199600
00199700
00199800
00199900
00200000
00200100
00200200
00200300
00200400
00200500
00200600
00200700
00200800
00200900
00201000
00201100
00201200
00201300
00201400
00201500
00201600
00201700
00201800
00201900
00202000

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

CALL ERROR ( 1004 )
GO TO 888
ENDIF
C
IF ( IRCA .EQ. 3 ) RETURN
IT = 0
IF ( BUFF(2:2) .NE. 1H ) IT = 2
IF ( BUFF(1:1) .NE. 1H ) IT = 1
C
BUFF(1:2) = 2H
C
IF ( IT .EQ. 1 ) CALL USMENT
IF ( IT .EQ. 2 ) CALL DBHENT
C
IF ( IT .EQ. 0 ) THEN
CALL ERROR( 1002 )
GO TO 200
END IF
C
GO TO 200
C
888 CONTINUE
999 RETURN
END

```

```

00202100
00202200
00202300
00202400
00202500
00202600
00202700
00202800
00202900
00203000
00203100
00203200
00203300
00203400
00203500
00203600
00203700
00203800
00203900
00204000
00204100
00204200
00204300
00204400

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE NUMSET *
C*****
SUBROUTINE NUMSET
C
C=====
C= FUNCTION =
C= TO SET THE /ADDR/ CHANGED NUMBER =
C=====
C-----
C- DATABASE MANAGEMENT AREA -
C-----
COMMON / MANAG / LRECOB, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LRECOB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE
C-----
C- ADDR FILES DATA -
C-----
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C-----
C- FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C
DO 200 I=1,ITNOW
IF ( IPRO(I) ) THEN
CALL CHANUM ( COND(IS(I):IE(I)), INUM(I) )
NCON = NCON + 1
END IF
200 CONTINUE
C
RETURN
END
    
```

```

00204500
00204600
00204700
00204800
00204900
00205000
00205100
00205200
00205300
00205400
00205500
00205600
00205700
00205800
00205900
00206000
00206100
00206200
00206300
00206400
00206500
00206600
00206700
00206800
00206900
00207000
00207100
00207200
00207300
00207400
00207500
00207600
00207700
00207800
00207900
00208000
00208100
00208200
00208300
00208400
00208500
00208600
00208700
00208800
00208900
00209000
00209100
00209200
    
```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C* SUBROUTINE P*W* *
C*****
SUBROUTINE P*W* ( TPW , IRC )
C
C=====
C= FUNCTION =
C= TO PUT THE USER-ID , USER PW & OWNER PW =
C= PARAMETER =
C= TPW ; TYPE IN PASSWORD (I) =
C= IRC ; RETURN ERROR CODE (O) =
C=====
C-----
C- PROCESS NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUS*OW, IMISS, IFUNC
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*,IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*,IPW
C
CHARACTER TPW*8, UPW*8, OPW*8
C
IRC = 0
C=
C= PASSWORD GENERATION
C=
CALL BAHAPW ( LOGID , UPW , OPW )
C
IF ( TPW(1:4) .EQ. '000?' ) THEN
WRITE ( ITSSOT , * ) LOGID , UPW , OPW
IRC = 1
RETURN
END IF
C
IF ( TPW(1:4) .EQ. '5963' .OR. TPW .EQ. OPW ) THEN
PASSWD = OPW
IUS*OW = 2
END IF
C
IF ( TPW(1:4) .EQ. '0000' .OR. TPW .EQ. UPW ) THEN
PASSWD = UPW
IUS*OW = 1
END IF
C
RETURN
    
```

```

00209300
00209400
00209500
00209600
00209700
00209800
00209900
00210000
00210100
00210200
00210300
00210400
00210500
00210600
00210700
00210800
00210900
00211000
00211100
00211200
00211300
00211400
00211500
00211600
00211700
00211800
00211900
00212000
00212100
00212200
00212300
00212400
00212500
00212600
00212700
00212800
00212900
00213000
00213100
00213200
00213300
00213400
00213500
00213600
00213700
00213800
00213900
00214000
00214100
00214200
00214300
00214400
00214500
00214600
00214700
    
```

.....1.....2.....3.....4.....5.....6.....7.....8

END

00214800

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C*  SUBROUTINE PAGE *
C*****
SUBROUTINE PAGE ( ITOT , IP , DT )
C
C-----
C=  FUNCTION =
C=    TO LISTUP THE PAGE & DATE =
C=  PARAMETER =
C=    ITOT ; TOTAL PAGES =
C=    IP ; LISTUP PAGE =
C=    DT ; TODAY'S DATE =
C-----
C
C-----
C-  UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+             ITSSIN, ITSSOT, IWORK*,IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+             IWORK*,IPW
C-----
C-  FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C
CHARACTER DT*8
C
WRITE ( ILIST , 1000 ) IP , ITOT , DT
C
1000 FORMAT ( //, 1H ,113X, 'PAGE :', 14, ' / ', 14,
+           /, 1H ,113X, 'DATE : ', A8 )
C
RETURN
END

```

00214900
00215000
00215100
00215200
00215300
00215400
00215500
00215600
00215700
00215800
00215900
00216000
00216100
00216200
00216300
00216400
00216500
00216600
00216700
00216800
00216900
00217000
00217100
00217200
00217300
00217400
00217500
00217600
00217700
00217800
00217900
00218000
00218100
00218200
00218300
00218400
00218500
00218600
00218700
00218800

PNC N9520 88-022(2)

- 98 -

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C  SUBROUTINE PROC *
C*****
SUBROUTINE PROC
C
C-----
C-  PROCCSS NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUSVOW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUSVOW, IMISS, IFUNC
C-----
C-  ADDR FILES DATA -
C-----
COMMON / ADDRFR / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C-----
C-  TERHTYPE COMMON AREA -
C-----
COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD
C
CHARACTER TNAME*8
C-----
C-  FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C-----
C-  PROCESS CONDITION AREA -
C-----
COMMON / CONDIT / JC(3)
C
INTEGER*2 JC
C-----
C-  ERROR MESSAGE AREA -
C-----
COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)
C
CHARACTER MFLAG*2, MSG*60
INTEGER*2 NMSG, NERCD
C-----
C-  RELATIVE ADDRESS AREA -

```

```

00218900
00219000
00219100
00219200
00219300
00219400
00219500
00219600
00219700
00219800
00219900
00220000
00220100
00220200
00220300
00220400
00220500
00220600
00220700
00220800
00220900
00221000
00221100
00221200
00221300
00221400
00221500
00221600
00221700
00221800
00221900
00222000
00222100
00222200
00222300
00222400
00222500
00222600
00222700
00222800
00222900
00223000
00223100
00223200
00223300
00223400
00223500
00223600
00223700
00223800
00223900
00224000
00224100
00224200
00224300

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C-----
COMMON / ISOU / ISO
C
INTEGER*4 ISO
C-----
C-  PICTURE COMMON AREA -
C-----
COMMON / PICT / MIA, PIA, LSIA(32)
C
CHARACTER MIA*16
C = = BUFF CLEAR
CALL BUFFCL
C
JC(1) = 0
100 CONTINUE
C
C = = TO DISPLAY THE PROC
IF( ITYPE .EQ. 1 ) THEN
MIA = 'SYSTEMU PROC'
IF( IMISS .EQ. 1 ) BUFF(001:010) = 'E0000G0000'
IF( IMISS .EQ. 2 ) BUFF(001:010) = '00070200'
IF( IMISS .EQ. 3 ) THEN
IF( JC(1) .EQ. 2 ) THEN
BUFF(001:010) = '00000G0000'
ELSE
BUFF(001:010) = '00000200'
ENDIF
ENDIF
IF( IMISS .EQ. 4 ) THEN
IF( JC(1) .EQ. 2 ) THEN
BUFF(001:010) = '00000G0000'
ELSE
BUFF(001:010) = '00000200'
ENDIF
ENDIF
ELSE
MIA = 'SYSTEMU PROC*'
IF( IMISS .EQ. 1 ) BUFF(001:010) = 'INPUT DATA'
IF( IMISS .EQ. 2 ) BUFF(001:010) = 'SEARCH KEY'
IF( IMISS .EQ. 3 ) THEN
IF( JC(1) .EQ. 2 ) THEN
BUFF(001:010) = 'INPUT DATA'
ELSE
BUFF(001:010) = 'UPDATE KEY'
ENDIF
ENDIF
IF( IMISS .EQ. 4 ) THEN
IF( JC(1) .EQ. 2 ) THEN
BUFF(001:010) = 'DEL DATA'
ELSE
BUFF(001:010) = 'DELETE KEY'
ENDIF
ENDIF
ENDIF

```

```

00224400
00224500
00224600
00224700
00224800
00224900
00225000
00225100
00225200
00225300
00225400
00225500
00225600
00225700
00225800
00225900
00226000
00226100
00226200
00226300
00226400
00226500
00226600
00226700
00226800
00226900
00227000
00227100
00227200
00227300
00227400
00227500
00227600
00227700
00227800
00227900
00228000
00228100
00228200
00228300
00228400
00228500
00228600
00228700
00228800
00228900
00229000
00229100
00229200
00229300
00229400
00229500
00229600
00229700
00229800

```


.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C*  SUBROUTINE RECCHK *
C*****
SUBROUTINE RECCHK
C
=====
C=  FUNCTION =
C=  TO CHECK THE DB RECORD & CONDITION RECORD =
C=====
C-----
C-  DATABASE MANAGEMENT AREA -
C-----
COMMON / MANAG / LRECD, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMP, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMP, IDATAS, IDATAE
C-----
C-  ADDR FILES DATA -
C-----
COMMON / ADDR / NDBAD(32000), HITAD(16,32000),
+ NMATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, HITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C-----
C-  UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, INSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*, IPW
C
INTEGER*2 IDB, IADDR, ILOG, INSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*, IPW
C-----
C-  FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C
COMMON / MLEVEL / LEVEL(16)
C
LOGICAL*1 LEVEL
C
IF ( NDOM .EQ. LRNOW ) GO TO 999
C=
C=  RECORD CHECK START

```

```

00236900
00237000
00237100
00237200
00237300
00237400
00237500
00237600
00237700
00237800
00237900
00238000
00238100
00238200
00238300
00238400
00238500
00238600
00238700
00238800
00238900
00239000
00239100
00239200
00239300
00239400
00239500
00239600
00239700
00239800
00239900
00240000
00240100
00240200
00240300
00240400
00240500
00240600
00240700
00240800
00240900
00241000
00241100
00241200
00241300
00241400
00241500
00241600
00241700
00241800
00241900
00242000
00242100
00242200
00242300

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C=
DO 200 I=1,LRNOW
IF ( NMATCH(I) .NE. NCON+NKEY ) GO TO 200
CDS
* WRITE(6,*) 'RECCHK NMATCH NCON',NMATCH(I),NCON
* WRITE(6,*) ' RECCHK ; NMATCH NCON I NDBAD(I) ',NMATCH(I),NCON,
+ I,NDBAD(I)
CDE
READ ( IDB , '(A)' , REC=NDBAD(I) ) BUFF*W(20)
CDS
* WRITE(6,*) ' RECCHK I K COND ',I,K,COND
* WRITE(6,*) ' RECCHK I K BUFF*W',I,K,BUFF*W(20)
CDE
J = 0
J2= 0
DO 100 K=1,ITNOW
IF ( .NOT.LEVEL(K) .AND. IPRO(K) .AND.
COND(IS(K):IE(K)) .EQ. BUFF*W(20)(IS(K):IE(K)) ) THEN
J = J + 1
C
ELSE IF ( LEVEL(K) .AND. IPRO(K) ) THEN
DO 50 L=1,ITNOW
IF ( LEVEL(L) .AND.
L.NE.K .AND.
BUFF*W(20)(IS(L):IE(L)) .NE.
20H .AND.
COND(IS(K):IE(K)) .EQ. BUFF*W(20)(IS(L):IE(L)) )
J2= J2+ 1
50 CONTINUE
END IF
C
100 CONTINUE
CDS
* WRITE(6,*) ' RECCHK J NMATCH',J,NMATCH(I)
CDE
IF ( J .EQ. NCON .AND.
+ ( NKEY .NE. 0 .AND. J2 .GE. 1 .OR.
+ NKEY .EQ. 0 ) ) THEN
NMATCH(I) = NCON + NKEY
ELSE
NMATCH(I) = 0
NDOM = NDOM - 1
CDS
* WRITE(6,*) ' RECCHK NMATCH,NDOM',NMATCH(I),NDOM
CDE
END IF
200 CONTINUE
C
999 CONTINUE
RETURN
END

```

```

00242400
00242500
00242600
00242700
00242800
00242900
00243000
00243100
00243200
00243300
00243400
00243500
00243600
00243700
00243800
00243900
00244000
00244100
00244200
00244300
00244400
00244500
00244600
00244700
00244800
00244900
00245000
00245100
00245200
00245300
00245400
00245500
00245600
00245700
00245800
00245900
00246000
00246100
00246200
00246300
00246400
00246500
00246600
00246700
00246800
00246900
00247000
00247100
00247200
00247300
00247400

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*** ALLOCATE DA('Q401#.BAHAMA.PW') F(FT10F001) MOD REU
CHARACTER U(2)=4,D(2)=4,D(2)=4
INTEGER IU(2),IO(2),ID(2)
EQUIVALENCE (U,IU),(O,IO),(D,ID)
100 CONTINUE
WRITE(6,*) ' USER-ID ? '
READ (5,'(2(A4))',END=9999) D(1),D(2)
WRITE(6,*) ' USER-PW ? '
READ (5,'(2(A4))') U(1),U(2)
WRITE(6,*) ' SYS -PW ? '
READ (5,'(2(A4))') O(1),O(2)
IU(1)=NOT(IU(1))
IU(2)=NOT(IU(2))
IO(1)=NOT(IO(1))
IO(2)=NOT(IO(2))
ID(1)=NOT(ID(1))
ID(2)=NOT(ID(2))
WRITE(10,'(6(A4))') D(1),D(2),U(1),U(2),O(1),O(2)
C
BACKSPACE 10
READ (10,'(6(A4))') D(1),D(2),U(1),U(2),O(1),O(2)
IU(1)=NOT(IU(1))
IU(2)=NOT(IU(2))
IO(1)=NOT(IO(1))
IO(2)=NOT(IO(2))
ID(1)=NOT(ID(1))
ID(2)=NOT(ID(2))
WRITE (6,*) D(1),D(2),U(1),U(2),O(1),O(2)
GO TO 100
9999 STOP
END
00247500
00247600
00247700
00247800
00247900
00248000
00248100
00248200
00248300
00248400
00248500
00248600
00248700
00248800
00248900
00249000
00249100
00249200
00249300
00249400
00249500
00249600
00249700
00249800
00249900
00250000
00250100
00250200
00250300
00250400
00250500

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C=> ALLOCATE THE DB TO FT10F001 & PS TO FT20F001
C=> DB:'Q401#.BAHAMA.DB' PS:WORK FILE LRECL=192
CHARACTER A(2)*192
WRITE(6,*) ' TYPE IN MODE ( FT10:DB , FT20:PS ) '
WRITE(6,*) ' 1:DB=>PS '
WRITE(6,*) ' 2:PS=>DB '
READ ( 5,* ) MODE
IF ( MODE .EQ. 1 ) THEN
100 CONTINUE
READ (10,'(2A192)',END=999) A(1),A(2)
WRITE(20,'(A192)') A(1)
WRITE(20,'(A192)') A(2)
GO TO 100
END IF
C
IF ( MODE .EQ. 2 ) THEN
200 CONTINUE
READ (20,'(A192)',END=999) A(1)
READ (20,'(A192)',END=999) A(2)
WRITE(10,'(2A192)') A(1),A(2)
GO TO 200
END IF
C
999 CONTINUE
STOP
END
00250600
00250700
00250800
00250900
00251000
00251100
00251200
00251300
00251400
00251500
00251600
00251700
00251800
00251900
00252000
00252100
00252200
00252300
00252400
00252500
00252600
00252700
00252800
00252900
00253000
00253100

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C  SUBROUTINE SEARCH *
C*****
C  SUBROUTINE SEARCH
C
C-----
C-  PROCCSS NOW CONDIION -
C-----
C  COMMON / PROSS / LOGID, PDATE, IUSWOW, PASSWD, MISS, IFUNC
C
C  CHARACTER LOGID*8, PDATE*8, PASSWD*8
C  INTEGER*2 IUSWOW, MISS, IFUNC
C-----
C-  ADDR FILES DATA -
C-----
C  COMMON / ADDR*8 / NDBAD(32000), HITAD(16,32000),
C  + NMATCH(32000),NCON, NDOM,
C  + ISEL(32000), IPRO(16), INUM(16),
C  + IS(16), IE(16), NKEY
C
C  INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
C  LOGICAL*1 ISEL, IPRO
C-----
C-  PROCESS CONDITION AREA -
C-----
C  COMMON / CONDIT / JC(3)
C
C  INTEGER*2 JC
C-----
C-  FILE INPUT/OUTPUT BUFFER -
C-----
C  COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
C  CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C-----
C-  ERROR MESSAGE AREA -
C-----
C  COMMON / ERRORM / MSG, MFLAG, NERCD(50), MSG(50)
C-----
C-  RELATIVE ADDRESS AREA -
C-----
C  COMMON / ISOU / ISO
C
C  INTEGER*4 ISO
C
C  CHARACTER MFLAG*2, MSG*60
C  INTEGER*2 NMSG, NERCD
C
C  IF( BUFF(225:249) .NE. 25H .OR.
C  + BUFF(250:274) .NE. 25H ) THEN

```

```

00253200
00253300
00253400
00253500
00253600
00253700
00253800
00253900
00254000
00254100
00254200
00254300
00254400
00254500
00254600
00254700
00254800
00254900
00255000
00255100
00255200
00255300
00255400
00255500
00255600
00255700
00255800
00255900
00256000
00256100
00256200
00256300
00256400
00256500
00256600
00256700
00256800
00256900
00257000
00257100
00257200
00257300
00257400
00257500
00257600
00257700
00257800
00257900
00258000
00258100
00258200
00258300
00258400
00258500
00258600

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

CALL COMAND (BUFF(225:274))
BUFF*W(19)(304:304) = '0'
BUFF(225:249) = 25H
BUFF(250:274) = 25H
GOTO 999
ENDIF
DD 100 I = 0 , 360 , 24
COND(I+1:I+24) = 24H
100 CONTINUE
IF( JC(1) .EQ. 0 ) THEN
COND(IS(1):IE(2)) = BUFF(11:74)
COND(IS(7):IE(12)) = BUFF(75:224)
BUFF*W(19)(1:384) = BUFF(1:384)
ELSE
COND(IS(1):IE(2)) = BUFF*W(19)(11:74)
COND(IS(7):IE(12)) = BUFF*W(19)(75:224)
ENDIF
C
C = SEARCH THE DB
CALL MATCHS
CALL MATCHC
C * = VERSION UP
***** IF( IUSWOW .NE. 2 ) CALL RECCHK
C * = VERSION UP
C
C = MATCH RECORD CHECK
IF( NDOM .LE. 0 ) THEN
IF( JC(1) .EQ. 2 ) THEN
CALL ERROR ( 2207 )
BUFF*W(19)(304:304) = '3'
JC(1) = 0
ELSE
CALL ERROR ( 2202 )
BUFF*W(19)(304:304) = '0'
ENDIF
GOTO 999
ENDIF
C
CALL SEL1
C
999 RETURN
END

```

```

00258700
00258800
00258900
00259000
00259100
00259200
00259300
00259400
00259500
00259600
00259700
00259800
00259900
00260000
00260100
00260200
00260300
00260400
00260500
00260600
00260700
00260800
00260900
00261000
00261100
00261200
00261300
00261400
00261500
00261600
00261700
00261800
00261900
00262000
00262100
00262200
00262300
00262400
00262500
00262600
00262700
00262800

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*****
C SUBROUTINE SELECT *
C*****
SUBROUTINE SELECT
C
C-----
C- PROCCSS NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUSVOW, PASSWD, IMISS, IFUNC
C
C CHARACTER LOGID*8, PDATE*8, PASSWD*8
C INTEGER*2 IUSVOW, IMISS, IFUNC
C-----
C- TERMTYPE COMMON AREA -
C-----
COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD
C
C CHARACTER TNAME*8
C-----
C- FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
C CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C-----
C- ERROR MESSAGE AREA -
C-----
COMMON / ERRORM / NMSG, HFLAG, NERCD(50), MSG(50)
C
C CHARACTER HFLAG*2, MSG*60
C INTEGER*2 NMSG, NERCD
C-----
C- PICTURE COMMON AREA -
C-----
COMMON / PICT / MIA, PIA, LSIA(32)
C
C CHARACTER MIA*16
C
100 CONTINUE
C
C IMISS = 0
C
C = = TO CLEAR THE DISPLAY AREA
CALL BUFFCL
C
C = = TO DISPLAY THE MENU.
IF( ITYPE .EQ. 1 ) THEN
MIA = 'SYSTEMU SELECT '
ELSE
MIA = 'SYSTEMU SELECT* '
ENDIF
00262900
00263000
00263100
00263200
00263300
00263400
00263500
00263600
00263700
00263800
00263900
00264000
00264100
00264200
00264300
00264400
00264500
00264600
00264700
00264800
00264900
00265000
00265100
00265200
00265300
00265400
00265500
00265600
00265700
00265800
00265900
00266000
00266100
00266200
00266300
00266400
00266500
00266600
00266700
00266800
00266900
00267000
00267100
00267200
00267300
00267400
00267500
00267600
00267700
00267800
00267900
00268000
00268100
00268200
00268300

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
CALL IPFMIO ( LSIA, MIA, BUFF, IRCA )
C
IF( IRCA .GT. 24 ) THEN
CALL ERROR ( 2001 )
GOTO 999
ENDIF
IF( IRCA .EQ. 3 ) GOTO 999
IF( IRCA .EQ. 4 ) GOTO 999
C
IF( BUFF(1:1) .NE. 1H .AND. BUFF(2:5) .EQ. 4H ) IMISS = 1
IF( BUFF(1:1) .EQ. 1H .AND. BUFF(2:2) .NE. 1H .AND.
+ BUFF(3:5) .EQ. 3H ) IMISS = 3
IF( BUFF(1:2) .EQ. 2H .AND. BUFF(3:3) .NE. 1H .AND.
+ BUFF(4:5) .EQ. 2H ) IMISS = 2
IF( BUFF(1:3) .EQ. 3H .AND. BUFF(4:4) .NE. 1H .AND.
+ BUFF(5:5) .EQ. 1H ) IMISS = 4
IF( BUFF(1:4) .EQ. 4H .AND. BUFF(5:5) .NE. 1H ) GOTO 999
C
IF( IMISS .EQ. 0 ) GOTO 100
C
CALL PROC
C
IF( BUFF*W(19)(301:302) .EQ. '03' ) GOTO 100
C
999 RETURN
END
00268400
00268500
00268600
00268700
00268800
00268900
00269000
00269100
00269200
00269300
00269400
00269500
00269600
00269700
00269800
00269900
00270000
00270100
00270200
00270300
00270400
00270500
00270600
00270700
00270800
00270900

```

DATA SET NAME : '0402#.AEG.FORT77(BAHAMA)'

DATE : 88/10/25

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*****
C SUBROUTINE SELECT1 * 00271000
C***** 00271100
SUBROUTINE SEL1 00271200
C 00271300
C----- 00271400
C- PROCCSS NOW CONDITION - 00271500
C----- 00271600
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, INISS, IFUNC 00271700
C 00271800
CHARACTER LOGID*8, PDATE*8, PASSWD*8 00271900
INTEGER*2 IUS*OW, INISS, IFUNC 00272000
C----- 00272100
C- DATABASE MANAGEMENT AREA - 00272200
C----- 00272300
COMMON / MANAG / LRECOB, LRMAX, LRNOW, ITMAX, ITNOW, 00272400
+ ITNAME(16), ITFORM(16), ITCHAR(16), 00272500
+ ITEMPS, ITEMP, IDATAS, IDATAE, SYSID 00272600
C 00272700
CHARACTER ITNAME*24, ITFORM*24, SYSID*5 00272800
INTEGER*2 LRECOB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS, 00272900
+ ITEMP, IDATAS, IDATAE 00273000
C----- 00273100
C- ADDR FILES DATA - 00273200
C----- 00273300
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000), 00273400
+ NMATCH(32000), NCON, NDOM, 00273500
+ ISEL(32000), IPRO(16), INUM(16), 00273600
+ IS(16), IE(16), NKEY 00273700
C 00273800
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY 00273900
LOGICAL*1 ISEL, IPRO 00274000
C----- 00274100
C- TERMTYPE COMMON AREA - 00274200
C----- 00274300
COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD 00274400
C 00274500
CHARACTER TNAME*8 00274600
C----- 00274700
C- FILE INPUT/OUTPUT BUFFER - 00274800
C----- 00274900
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL 00275000
C 00275100
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384 00275200
C----- 00275300
C- PROCESS CONDITION AREA - 00275400
C----- 00275500
COMMON / CONDIR / JC(3) 00275600
C 00275700
INTEGER*2 JC 00275800
00275900
00276000
00276100
00276200
00276300
00276400

```

DATA SET NAME : '0402#.AEG.FORT77(BAHAMA)'

```

.....1.....2.....3.....4.....5.....6.....7.....8
C 00276500
C----- 00276600
C- ERROR MESSAGE AREA - 00276700
C----- 00276800
COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50) 00276900
C 00277000
CHARACTER MFLAG*2, MSG*60 00277100
INTEGER*2 NMSG, NERCD 00277200
C----- 00277300
C- PICTURE COMMON AREA - 00277400
C----- 00277500
COMMON / PICT / MIA, PIA, LSIA(32) 00277600
C 00277700
CHARACTER MIA*16 00277800
C----- 00277900
C- GETDB COMMON AREA - 00278000
C----- 00278100
COMMON / GEDB / ISTART, IDIM, IEND, ICO, IRAD 00278200
C 00278300
INTEGER *4 IRAD(20) 00278400
C----- 00278500
C- RELATIVE ADDRESS AREA - 00278600
C----- 00278700
COMMON / ISOU / ISO 00278800
C 00278900
INTEGER*4 ISO 00279000
C----- 00279100
C- SEL1 USED AREA - 00279200
C----- 00279300
INTEGER *4 IPAGE1, IPAGE2 00279400
C 00279500
C = = DATA SET 00279600
IPAGE1 = 0 00279700
ISTART = 1 00279800
IDIM = +1 00279900
C 00280000
C = = BUFF AREA SPACE CLEAR 00280100
DO 100 I = 0, 1200, 100 00280200
BUFF(I+1:I+100) = 100H 00280300
+ 00280400
100 CONTINUE 00280500
BUFF(1301:1312) = 12H 00280600
C 00280700
IPAGE2 = ( NDOM - 1 ) / 18 + 1 00280800
C 00280900
200 CALL GETDB ( ISTART, IDIM, IEND, ICO, IRAD, IER ) 00281000
C 00281100
IF( IER .EQ. 1 .OR. IER .EQ. 2 ) GOTO 500 00281200
C 00281300
IF( ICO .EQ. 0 ) THEN 00281400
IF( JC(1) .EQ. 0 ) THEN 00281500
00281600
00281700
00281800
00281900

```

DATE : 88/10/25

```

.....1.....2.....3.....4.....5.....6.....7.....8
      CALL ERROR ( 2202 )
      BUFF*W(19)(304:304) = '0'
      GOTO 999
    ELSE
      GOTO 500
    ENDIF
  ENDIF
C
  IF( IDIM .EQ. 1 ) THEN
    IPAGE1 = IPAGE1 + 1
  ELSE
    IPAGE1 = IPAGE1 - 1
  ENDIF
  WRITE(BUFF(0011:0014),'(I4)') IPAGE1
  WRITE(BUFF(0015:0018),'(I4)') IPAGE2
C
300 IF( ITYPE .EQ. 1 ) THEN
  IF( IMISS .EQ. 2 ) BUFF(0001:0010) = '00070000 '
  IF( IMISS .EQ. 3 ) BUFF(0001:0010) = '060000A0B2'
  IF( IMISS .EQ. 4 ) BUFF(0001:0010) = '060000A0B2'
  MIA = 'SYSTEMU SEL1 '
  ELSE
  IF( IMISS .EQ. 2 ) BUFF(0001:0010) = 'SEARCH RES'
  IF( IMISS .EQ. 3 ) BUFF(0001:0010) = ' DATA SEL '
  IF( IMISS .EQ. 4 ) BUFF(0001:0010) = ' DATA SEL '
  MIA = 'SYSTEMU SEL1* '
  ENDIF
C
  I3 = 19
  DO 350 I = 1 , 18
    BUFF(I3:I3+64) = 65H
  +
    I3 = I3 + 65
  350 CONTINUE
C
  I3 = 19
  J = ( IPAGE1 - 1 ) * 18
  DO 400 I = 1 , ICO
    BUFF(I3+1:I3+44) = BUFF*W(I)(001:044)
    BUFF(I3+45:I3+64) = BUFF*W(I)(045:064)
    IF( IRAD(I) .EQ. ISO ) BUFF(I3:I3) = 'S'
    I3 = I3 + 65
  400 CONTINUE
C
  500 CALL IPFM10 ( LSIA, MIA, BUFF, IRCA )
C
C = = RETURN CODE CHECK
  DO 550 I = 301 , 384 , 1
    BUFF*W(19)(I:1) = '0'
  550 CONTINUE
C
  IF( IRCA .GT. 24 ) THEN
    CALL ERROR ( 2203 )
    GOTO 999
  ENDIF
00282000
00282100
00282200
00282300
00282400
00282500
00282600
00282700
00282800
00282900
00283000
00283100
00283200
00283300
00283400
00283500
00283600
00283700
00283800
00283900
00284000
00284100
00284200
00284300
00284400
00284500
00284600
00284700
00284800
00284900
00285000
00285100
00285200
00285300
00285400
00285500
00285600
00285700
00285800
00285900
00286000
00286100
00286200
00286300
00286400
00286500
00286600
00286700
00286800
00286900
00287000
00287100
00287200
00287300
00287400

```

DATA SET NAME : 'Q402#.AEG.FORT77(BAHAMA)'

```

.....1.....2.....3.....4.....5.....6.....7.....8
C
  IF( IRCA .EQ. 0 ) THEN
    JC(1) = 2
  ENDIF
C
C = = SELECT FLAG CHECK
  IF( IMISS .EQ. 3 .OR. IMISS .EQ. 4 ) THEN
    J1 = 19
    ** = 0
    DO 600 I = 1 , ICO
      IF( BUFF(J1:J1) .NE. 1H ) THEN
        IF( ** .EQ. 0 ) THEN
          CSEL = BUFF*W(I)
          ** = 1
        ENDIF
        IF( IRCA .EQ. 0 ) THEN
          ISO = IRAD(I)
          BUFF*W(19)(304:304) = '3'
          GOTO 999
        ENDIF
        ENDIF
        J1 = J1 + 65
    600 CONTINUE
  ENDIF
C
  IF( IRCA .EQ. 1 ) THEN
    BUFF*W(19)(304:304) = '1'
    GOTO 999
  ENDIF
  IF( IRCA .EQ. 3 ) THEN
    BUFF*W(19)(304:304) = '3'
    JC(1) = 0
    GOTO 999
  ENDIF
  IF( IRCA .EQ. 4 ) THEN
    BUFF*W(19)(304:304) = '4'
    GOTO 999
  ENDIF
  IF( IRCA .EQ. 7 ) THEN
    IF( (IPAGE1-1)*18+ICO .LE. 18 ) GOTO 500
    IF( IDIM .EQ. -1 ) ISTART = IEND
    ISTART = ISTART - 1
    IDIM = -1
    JC(3) = 1
    GOTO 200
  ENDIF
  IF( IRCA .EQ. 8 ) THEN
    IF( NDOH .LE. (IPAGE1-1)*18+ICO ) GOTO 500
    IF( IDIM .EQ. -1 ) IEND = ISTART
    ISTART = IEND + 1
    IDIM = +1
    JC(3) = 1
    GOTO 200
  ENDIF
  IF( IRCA .EQ. 11 ) THEN
00287500
00287600
00287700
00287800
00287900
00288000
00288100
00288200
00288300
00288400
00288500
00288600
00288700
00288800
00288900
00289000
00289100
00289200
00289300
00289400
00289500
00289600
00289700
00289800
00289900
00290000
00290100
00290200
00290300
00290400
00290500
00290600
00290700
00290800
00290900
00291000
00291100
00291200
00291300
00291400
00291500
00291600
00291700
00291800
00291900
00292000
00292100
00292200
00292300
00292400
00292500
00292600
00292700
00292800
00292900

```

DATE : 88/10/25

```

.....1.....2.....3.....4.....5.....6.....7.....8
      CALL SEL2
      IF( BUFF*W(19)(304:304) .EQ. '1' .OR.
+      BUFF*W(19)(304:304) .EQ. '3' .OR.
+      BUFF*W(19)(304:304) .EQ. '4' ) GOTO 999
      READ(BUFF*W(19)(311:314),'(8N,I4)') IPAGE1
      GOTO 300
    ENDIF
    IF( IRCA .EQ. 12 ) CALL LISTUP ( NERR )
  C
  GOTO 300
  C
  999 CONTINUE
  DO 1000 I = 18 , 1278 , 30
    BUFF(I+1:I+30) = 30H
  1000 CONTINUE
  RETURN
  END

```

```

00293000
00293100
00293200
00293300
00293400
00293500
00293600
00293700
00293800
00293900
00294000
00294100
00294200
00294300
00294400
00294500
00294600

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*****
C  SUBROUTINE SELECT2 *
C*****
SUBROUTINE SEL2
C
C-----
C-  PROCESS NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUS*OW, IMISS, IFUNC
C
C-----
C-  ADDR FILES DATA -
C-----
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000),NCON, NDOM,
+ ISEL(32000), IPRD(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRD
C
C-----
C-  TERNTYPE COMMON AREA -
C-----
COMMON / TERM / IRO , IR1 , TNAME , ITYPE , IRCD
C
CHARACTER TNAME*8
C
C-----
C-  FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C
C-----
C-  PROCESS CONDITION AREA -
C-----
COMMON / CONDIT / JC(3)
C
INTEGER*2 JC
C
C-----
C-  ERROR MESSAGE AREA -
C-----
COMMON / ERRORH / NMSG, MFLAG, NERCD(50), MSG(50)
C
CHARACTER MFLAG*2, HSG*60
INTEGER*2 NMSG, NERCD
C
C-----
C-  PICTURE COMMON AREA -

```

```

00294700
00294800
00294900
00295000
00295100
00295200
00295300
00295400
00295500
00295600
00295700
00295800
00295900
00296000
00296100
00296200
00296300
00296400
00296500
00296600
00296700
00296800
00296900
00297000
00297100
00297200
00297300
00297400
00297500
00297600
00297700
00297800
00297900
00298000
00298100
00298200
00298300
00298400
00298500
00298600
00298700
00298800
00298900
00299000
00299100
00299200
00299300
00299400
00299500
00299600
00299700
00299800
00299900
00300000
00300100

```



```

.....1.....2.....3.....4.....5.....6.....7.....8
C-----
COMMON / PICT / MIA , PIA , LSIA(32)
C
CHARACTER MIA*16
C-----
C- GETDB COMMON AREA
C-----
COMMON / GEDB / ISTART , IDIM , IEND , ICO , IRAD
C
C = DATA SET
100 CONTINUE
C
DO 200 I = 18 , 1278 , 30
    BUFF(I+1:I+30) = 30H
200 CONTINUE
C
    BUFF(1309:1312) = 4H
C
READ(BUFF(0011:0014),'(BN,14)') IPAGE1
GOTO 400
C
300 CONTINUE
C
CALL GETDB ( ISTART , IDIM , IEND , ICO , IRAD , IER )
IF( IER .EQ. 1 .OR. IER .EQ. 2 ) GOTO 600
IF( IDIM .EQ. 1 ) THEN
    IPAGE1 = IPAGE1 + 1
ELSE
    IPAGE1 = IPAGE1 - 1
ENDIF
WRITE(BUFF(0011:0014),'(14)') IPAGE1
C
400 CONTINUE
C
I3 = 19
DO 450 I = 1 , 18
    BUFF(13:13+4) = 5H
    BUFF(13+5:13+12) = 8H
    BUFF(13+13:13+17) = 5H
    BUFF(13+18:13+25) = 8H
    BUFF(13+26:13+45) = 20H
    BUFF(13+46:13+65) = 20H
    I3 = I3 + 66
450 CONTINUE
I3 = 19
DO 500 I = 1 , ICO
    BUFF(13:13+4) = BUFF*W(1)(065:069)
    BUFF(13+5:13+12) = BUFF*W(1)(070:077)
    BUFF(13+13:13+17) = BUFF*W(1)(078:082)
    BUFF(13+18:13+25) = BUFF*W(1)(083:090)
    BUFF(13+26:13+45) = BUFF*W(1)(091:110)
    BUFF(13+46:13+65) = BUFF*W(1)(111:130)
    I3 = I3 + 66
500 CONTINUE

```

```

00300200
00300300
00300400
00300500
00300600
00300700
00300800
00300900
00301000
00301100
00301200
00301300
00301400
00301500
00301600
00301700
00301800
00301900
00302000
00302100
00302200
00302300
00302400
00302500
00302600
00302700
00302800
00302900
00303000
00303100
00303200
00303300
00303400
00303500
00303600
00303700
00303800
00303900
00304000
00304100
00304200
00304300
00304400
00304500
00304600
00304700
00304800
00304900
00305000
00305100
00305200
00305300
00305400
00305500
00305600

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C
600 CONTINUE
C
IF( ITYPE .EQ. 1 ) THEN
    MIA = 'SYSHENU SEL2 '
ELSE
    MIA = 'SYSHENU SEL2* '
ENDIF
C
CALL IPFNIO (LSIA, MIA, BUFF, IRCA )
C
C = RETURN CODE CHECK
DO 650 I = 301 , 384 , 1
    BUFF*W(19)(I:1) = '0'
650 CONTINUE
C
IF( IRCA .GT. 24 ) THEN
    CALL ERROR ( 2204 )
    GOTO 999
ENDIF
IF( IRCA .EQ. 1 ) THEN
    BUFF*W(19)(304:304) = '1'
    GOTO 999
ENDIF
IF( IRCA .EQ. 3 ) THEN
    BUFF*W(19)(304:304) = '3'
    GOTO 999
ENDIF
IF( IRCA .EQ. 4 ) THEN
    BUFF*W(19)(304:304) = '4'
    GOTO 999
ENDIF
IF( IRCA .EQ. 7 ) THEN
    IF((IPAGE1-1)*18+ICO .LE. 18 ) GOTO 600
    IF( IDIM .EQ. -1 ) ISTART = IEND
    ISTART = ISTART - 1
    IDIM = -1
    JC(3) = 1
    GOTO 300
ENDIF
IF( IRCA .EQ. 8 ) THEN
    IF( NDOH .LE. (IPAGE1-1)*18+ICO ) GOTO 600
    IF( IDIM .EQ. -1 ) IEND = ISTART
    ISTART = IEND + 1
    IDIM = +1
    JC(3) = 1
    GOTO 300
ENDIF
WRITE(BUFF*W(19)(311:314),'(14)') IPAGE1
IF( IRCA .EQ. 10 ) THEN
    BUFF*W(19)(303:303) = '1'
    GOTO 999
ENDIF
IF( IRCA .EQ. 11 ) THEN
    CALL SEL3

```

```

00305700
00305800
00305900
00306000
00306100
00306200
00306300
00306400
00306500
00306600
00306700
00306800
00306900
00307000
00307100
00307200
00307300
00307400
00307500
00307600
00307700
00307800
00307900
00308000
00308100
00308200
00308300
00308400
00308500
00308600
00308700
00308800
00308900
00309000
00309100
00309200
00309300
00309400
00309500
00309600
00309700
00309800
00309900
00310000
00310100
00310200
00310300
00310400
00310500
00310600
00310700
00310800
00310900
00311000
00311100

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
      IF( BUFF*W(19)(304:304) .EQ. '1' .OR.          00311200
+      BUFF*W(19)(304:304) .EQ. '3' .OR.          00311300
+      BUFF*W(19)(304:304) .EQ. '4') GOTO 999      00311400
      READ(BUFF*W(19)(311:314),'(BN,I4)') IPAGE1    00311500
      ENDIF                                         00311600
      IF( IRCA .EQ. 12 ) CALL LISTUP ( NERR )       00311700
C
      GOTO 100                                       00311800
C
      RETURN                                         00311900
999      END                                         00312000
          END                                         00312100
          END                                         00312200
  
```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C*****
C  SUBROUTINE SELECT3                               *          00312300
C*****
C  SUBROUTINE SEL3                                 *          00312400
C*****
C  SUBROUTINE SEL3                                 *          00312500
C*****
C-----
C-  PROCESS NOW CONDITION                          -          00312600
C-----
C  COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC 00312700
C
C  CHARACTER LOGID*8, PDATE*8, PASSWD*8           00312800
C  INTEGER*2 IUS*OW, IMISS, IFUNC                 00312900
C-----
C-  ADDR FILES DATA                              -          00313000
C-----
C  COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000), 00313100
C  + NMATCH(32000), NCON, NDOM,                   00313200
C  + ISEL(32000), IPRO(16), INUM(16),             00313300
C  + IS(16), IE(16), NKEY                          00313400
C
C  INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY 00313500
C  LOGICAL*1 ISEL, IPRO                             00313600
C-----
C-  TERM TYPE COMMON AREA                          -          00313700
C-----
C  COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD    00313800
C
C  CHARACTER TNAME*8                               00313900
C-----
C-  FILE INPUT/OUTPUT BUFFER                       -          00314000
C-----
C  COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL  00314100
C
C  CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384 00314200
C-----
C-  PROCESS CONDITION AREA                          -          00314300
C-----
C  COMMON / CONDIT / JC(3)                         00314400
C
C  INTEGER*2 JC                                    00314500
C-----
C-  ERROR MESSAGE AREA                             -          00314600
C-----
C  COMMON / ERRORH / NMSG, HFLAG, NERCD(50), MSG(50) 00314700
C
C  CHARACTER HFLAG*2, MSG*60                       00314800
C  INTEGER*2 NMSG, NERCD                           00314900
C-----
C-  PICTURE COMMON AREA                            -          00315000
  
```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C-----
COMMON / PICT / MIA , PIA , LSIA(32)
C
CHARACTER MIA*16
C-----
C- GETDB COMMON AREA
C-----
COMMON / GEDB / ISTART , IDIM , IEND , ICO , IRAD
C
100 CONTINUE
C
DO 200 I = 18 , 1278 , 30
  BUFF(1+1:I+30) = 30H
200 CONTINUE
C
  BUFF(1309:1312) = 4H
C
  READ(BUFF(0011:0014),'(BN,I4)') IPAGE1
  GOTO 400
C
300 CONTINUE
C
CALL GETDB ( ISTART , IDIM , IEND , ICO , IRAD , IER )
IF( IER .EQ. 1 .OR. IER .EQ. 2 ) GOTO 600
C
IF( IDIM .EQ. 1 ) THEN
  IPAGE1 = IPAGE1 + 1
ELSE
  IPAGE1 = IPAGE1 - 1
ENDIF
WRITE(BUFF(0011:0014),'(I4)') IPAGE1
C
400 CONTINUE
C
I3 = 19
DO 450 I = 1 , 18
  BUFF(I3:I3+19) = 20H
  BUFF(I3+20:I3+39) = 20H
  BUFF(I3+40:I3+59) = 20H
  I3 = I3 + 60
450 CONTINUE
I3 = 19
DO 500 I = 1 , ICO
  BUFF(I3:I3+19) = BUFF*W(I)(131:150)
  BUFF(I3+20:I3+39) = BUFF*W(I)(151:170)
  BUFF(I3+40:I3+59) = BUFF*W(I)(171:190)
  I3 = I3 + 60
500 CONTINUE
C
600 CONTINUE
C
IF( ITYPE .EQ. 1 ) THEN
  MIA = 'SYSMENU SEL3'
ELSE

```

```

00317800
00317900
00318000
00318100
00318200
00318300
00318400
00318500
00318600
00318700
00318800
00318900
00319000
00319100
00319200
00319300
00319400
00319500
00319600
00319700
00319800
00319900
00320000
00320100
00320200
00320300
00320400
00320500
00320600
00320700
00320800
00320900
00321000
00321100
00321200
00321300
00321400
00321500
00321600
00321700
00321800
00321900
00322000
00322100
00322200
00322300
00322400
00322500
00322600
00322700
00322800
00322900
00323000
00323100
00323200

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

MIA = 'SYSMENU SEL3'
ENDIF
C
CALL IPFMIO ( LSIA , MIA , BUFF , IRCA )
C
C = RETURN CODE CHECK
DO 650 I = 301 , 384 , 1
  BUFF*W(19)(I:1) = '0'
650 CONTINUE
C
IF( IRCA .GT. 24 ) THEN
  CALL ERROR ( 2205 )
  GOTO 999
ENDIF
IF( IRCA .EQ. 1 ) THEN
  BUFF*W(19)(304:304) = '1'
  GOTO 999
ENDIF
IF( IRCA .EQ. 3 ) THEN
  BUFF*W(19)(304:304) = '3'
  GOTO 999
ENDIF
IF( IRCA .EQ. 4 ) THEN
  BUFF*W(19)(304:304) = '4'
  GOTO 999
ENDIF
IF( IRCA .EQ. 7 ) THEN
  IF((IPAGE1-1)*18+ICO .LE. 18 ) GOTO 600
  IF( IDIM .EQ. -1 ) ISTART = IEND
  ISTART = ISTART - 1
  IDIM = -1
  JC(3) = 1
  GOTO 300
ENDIF
IF( IRCA .EQ. 8 ) THEN
  IF( NDOM .LE. (IPAGE1-1)*18+ICO ) GOTO 600
  IF( IDIM .EQ. -1 ) IEND = ISTART
  ISTART = IEND + 1
  IDIM = +1
  JC(3) = 1
  GOTO 300
ENDIF
WRITE(BUFF*W(19)(311:314),'(I4)') IPAGE1
IF( IRCA .EQ. 10 ) THEN
  BUFF*W(19)(303:303) = '1'
  GOTO 999
ENDIF
IF( IRCA .EQ. 11 ) THEN
  CALL SEL4
  IF( BUFF*W(19)(304:304) .EQ. '1' .OR.
  + BUFF*W(19)(304:304) .EQ. '3' .OR.
  + BUFF*W(19)(304:304) .EQ. '4' ) GOTO 999
  READ(BUFF*W(19)(311:314),'(BN,I4)') IPAGE1
ENDIF
IF( IRCA .EQ. 12 ) CALL LISTUP ( NERR )

```

```

00323300
00323400
00323500
00323600
00323700
00323800
00323900
00324000
00324100
00324200
00324300
00324400
00324500
00324600
00324700
00324800
00324900
00325000
00325100
00325200
00325300
00325400
00325500
00325600
00325700
00325800
00325900
00326000
00326100
00326200
00326300
00326400
00326500
00326600
00326700
00326800
00326900
00327000
00327100
00327200
00327300
00327400
00327500
00327600
00327700
00327800
00327900
00328000
00328100
00328200
00328300
00328400
00328500
00328600
00328700

```

.....1.....2.....3.....4.....5.....6.....7.....8

C
GOTO 100
C
999 RETURN
END

00328800
00328900
00329000
00329100
00329200

.....1.....2.....3.....4.....5.....6.....7.....8

C*****
C SUBROUTINE SELECT4 *
C*****
SUBROUTINE SEL4
C
C-----
C- PROCESS NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUSVOW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*8
INTEGER*2 IUSVOW, IMISS, IFUNC
C
C-----
C- ADDR FILES DATA -
C-----
COMMON / ADDRFR / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C
C-----
C- TERNTYPE COMMON AREA -
C-----
COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD
C
CHARACTER TNAME*8
C
C-----
C- FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C
C-----
C- PROCESS CONDITION AREA -
C-----
COMMON / CONDIR / JC(3)
C
INTEGER*2 JC
C
C-----
C- ERROR MESSAGE AREA -
C-----
COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)
C
CHARACTER MFLAG*2, MSG*60
INTEGER*2 NMSG, NERCD
C
C-----
C- PICTURE COMMON AREA -

00329300
00329400
00329500
00329600
00329700
00329800
00329900
00330000
00330100
00330200
00330300
00330400
00330500
00330600
00330700
00330800
00330900
00331000
00331100
00331200
00331300
00331400
00331500
00331600
00331700
00331800
00331900
00332000
00332100
00332200
00332300
00332400
00332500
00332600
00332700
00332800
00332900
00333000
00333100
00333200
00333300
00333400
00333500
00333600
00333700
00333800
00333900
00334000
00334100
00334200
00334300
00334400
00334500
00334600
00334700

PNC N9520 88-022(2)

- 111 -

```

.....1.....2.....3.....4.....5.....6.....7.....8
C-----
COMMON / PICT / MIA , PIA , LSIA(32)
C
CHARACTER MIA*16
C-----
C- GETDB COMMON AREA
C-----
COMMON / GEDB / ISTART , IDIM , IEND , ICO , IRAD
C
100 CONTINUE
C
DO 200 I = 18 , 1278 , 30
  BUFF(I+1:I+30) = 30H
200 CONTINUE
C
  BUFF(1309:1312) = 4H
C
  READ(BUFF(0011:0014),'(BN,I4)') IPAGE1
  GOTO 400
C
300 CONTINUE
C
CALL GETDB ( ISTART , IDIM , IEND , ICO , IRAD , IER )
IF( IER .EQ. 1 .OR. IER .EQ. 2 ) GOTO 600
C
IF( IDIM .EQ. 1 ) THEN
  IPAGE1 = IPAGE1 + 1
ELSE
  IPAGE1 = IPAGE1 - 1
ENDIF
WRITE(BUFF(0011:0014),'(I4)') IPAGE1
C
400 CONTINUE
C
I3 = 19
DO 450 I = 1 , 18
  BUFF(I3:I3+49) = 50H
+
  I3 = I3 + 50
450 CONTINUE
I3 = 19
DO 500 I = 1 , ICO
  BUFF(I3:I3+49) = BUFF*W(I)(191:240)
  I3 = I3 + 50
500 CONTINUE
C
600 CONTINUE
C
IF( ITYPE .EQ. 1 ) THEN
  MIA = 'SYSTEMU SEL4 '
ELSE
  MIA = 'SYSTEMU SEL4Y '
ENDIF
C

```

```

00334800
00334900
00335000
00335100
00335200
00335300
00335400
00335500
00335600
00335700
00335800
00335900
00336000
00336100
00336200
00336300
00336400
00336500
00336600
00336700
00336800
00336900
00337000
00337100
00337200
00337300
00337400
00337500
00337600
00337700
00337800
00337900
00338000
00338100
00338200
00338300
00338400
00338500
00338600
00338700
00338800
00338900
00339000
00339100
00339200
00339300
00339400
00339500
00339600
00339700
00339800
00339900
00340000
00340100
00340200

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
C
CALL IPFMIO ( LSIA , MIA , BUFF , IRCA )
C
C = RETURN CODE CHECK
DO 650 I = 301 , 384 , 1
  BUFF*W(19)(I:1) = '0'
650 CONTINUE
C
IF( IRCA .GT. 24 ) THEN
  CALL ERRDR ( 2206 )
  GOTO 999
ENDIF
IF( IRCA .EQ. 1 ) THEN
  BUFF*W(19)(304:304) = '1'
  GOTO 999
ENDIF
IF( IRCA .EQ. 3 ) THEN
  BUFF*W(19)(304:304) = '3'
  GOTO 999
ENDIF
IF( IRCA .EQ. 4 ) THEN
  BUFF*W(19)(304:304) = '4'
  GOTO 999
ENDIF
IF( IRCA .EQ. 7 ) THEN
  IF( (IPAGE1-1)*18+ICO .LE. 18 ) GOTO 600
  IF( IDIM .EQ. -1 ) ISTART = IEND
  ISTART = ISTART - 1
  IDIM = -1
  JC(3) = 1
  GOTO 300
ENDIF
IF( IRCA .EQ. 8 ) THEN
  IF( NDOM .LE. (IPAGE1-1)*18+ICO ) GOTO 600
  IF( IDIM .EQ. -1 ) IEND = ISTART
  ISTART = IEND + 1
  IDIM = +1
  JC(3) = 1
  GOTO 300
ENDIF
IF( IRCA .EQ. 10 ) THEN
  WRITE(BUFF*W(19)(311:314),'(I4)') IPAGE1
  BUFF*W(19)(303:303) = '1'
  GOTO 999
ENDIF
IF( IRCA .EQ. 12 ) CALL LISTUP ( NERR )
C
GOTO 100
C
999 RETURN
END

```

```

00340300
00340400
00340500
00340600
00340700
00340800
00340900
00341000
00341100
00341200
00341300
00341400
00341500
00341600
00341700
00341800
00341900
00342000
00342100
00342200
00342300
00342400
00342500
00342600
00342700
00342800
00342900
00343000
00343100
00343200
00343300
00343400
00343500
00343600
00343700
00343800
00343900
00344000
00344100
00344200
00344300
00344400
00344500
00344600
00344700
00344800
00344900
00345000
00345100
00345200

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
SUBROUTINE UADD ( WORK )
C-----
C= FUNCTION =
C= TO ADD THE USER TO PASSWRD FILE =
C= PARAMETER =
C= WORK : TSS COMAND AREA =
C-----
C
C-----
C- PROCCES NOW CONDITION -
C-----
C COMMON / PROSS / LOGID, PDATE, IUSYOW, PASSWD, IMISS, IFUNC
C
C CHARACTER LOGID*8, PDATE*8, PASSWD*4
C INTEGER*2 IUSYOW, IMISS, IFUNC
C-----
C- DATABASE MANAGEMENT AREA -
C-----
C COMMON / MANAG / LRECD, LRMAX, LRNOW, ITMAX, ITNOW,
C + ITNAME(16), ITFORM(16), ITCHAR(16),
C + ITEMP, ITEMPE, IDATAS, IDATAE, SYSID
C
C CHARACTER ITNAME*24, ITFORM*24, SYSID*5
C INTEGER*2 LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMP,
C + ITEMPE, IDATAS, IDATAE
C-----
C- PROCESS CONDITION AREA -
C-----
C COMMON / CONdit / JC(3)
C
C INTEGER*2 JC
C-----
C- ERROR MESSAGE AREA -
C-----
C COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)
C
C CHARACTER MFLAG*2, MSG*60
C INTEGER*2 NMSG, NERCD
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
C COMMON / UNIT / IOB, IADDR, ILOG, IMSG, IWORK, ILIST,
C + ITSSIN, ITSSOT, IWORK*, IPW
C
C INTEGER*2 IOB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
C + IWORK*, IPW
C-----
C- ADDR FILES DATA -
C-----
C COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),

```

```

00345300
00345400
00345500
00345600
00345700
00345800
00345900
00346000
00346100
00346200
00346300
00346400
00346500
00346600
00346700
00346800
00346900
00347000
00347100
00347200
00347300
00347400
00347500
00347600
00347700
00347800
00347900
00348000
00348100
00348200
00348300
00348400
00348500
00348600
00348700
00348800
00348900
00349000
00349100
00349200
00349300
00349400
00349500
00349600
00349700
00349800
00349900
00350000
00350100
00350200
00350300
00350400
00350500
00350600
00350700

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
+ NMATCH(32000),NCON, NDOM, 00350800
+ ISEL(32000), IPRO(16), INUM(16), 00350900
+ IS(16), IE(16), NKEY 00351000
C 00351100
C INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY 00351200
C LOGICAL*1 ISEL, IPRO 00351300
C-----
C- DB FILE EMPTY RECORD CONTROL - 00351400
C-----
C COMMON / BITMAP / MAP 00351500
C 00351600
C CHARACTER MAP*4000 00351700
C-----
C 00351800
C 00351900
C 00352000
C-----
C- TERMTYPE COMMON AREA - 00352100
C-----
C 00352200
C COMMON / TERM / IRO, IRI, TNAME, ITYPE, IRCD 00352300
C 00352400
C CHARACTER TNAME*8 00352500
C-----
C 00352600
C-----
C- FILE INPUT/OUTPUT BUFFER - 00352700
C-----
C COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL 00352800
C 00352900
C CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384 00353000
C 00353100
C 00353200
C 00353300
C 00353400
C-----
C- PICTURE COMMON AREA - 00353500
C-----
C COMMON / PICT / MIA, PIA, LSIA(32) 00353600
C 00353700
C 00353800
C 00353900
C 00354000
C 00354100
C CHARACTER MIA*16 00354200
C 00354300
C CHARACTER U(2)*4, O(2)*4, D(2)*4, WORK(3)*50 00354400
C 00354500
C INTEGER IU(2), IO(2), ID(2)
C 00354600
C EQUIVALENCE (U,IU),(O,IO),(D,ID)
C 00354700
C==> START 00354800
C 00354900
C IF ( BUFF(02:09) .EQ. 8H ) THEN 00355000
C CALL ERROR ( 1012 ) 00355100
C GO TO 999 00355200
C END IF 00355300
C 00355400
C 00355500
C IF ( BUFF(10:17) .EQ. 8H .AND. 00355600
C + BUFF(18:25) .EQ. 8H ) THEN
C CALL ERROR ( 1013 ) 00355700
C GO TO 999 00355800
C END IF 00355900
C 00356000
C IF ( BUFF(10:17) .EQ. BUFF(18:25) ) THEN 00356100
C CALL ERROR ( 1017 ) 00356200

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
      GO TO 999                                00356300
      END IF                                    00356400
C
      CALL COMAND ( WORK(1) )                  00356500
      CALL COMAND ( WORK(2) )                  00356600
      OPEN ( 11 )                               00356700
C
      D(1) = BUFF(02:05)                       00356800
      D(2) = BUFF(06:06)///                    00356900
      U(1) = BUFF(10:13)                       00357000
      U(2) = BUFF(14:17)                       00357100
      O(1) = BUFF(18:21)                       00357200
      O(2) = BUFF(22:25)                       00357300
C
      IU(1)=NOT(IU(1))                          00357400
      IU(2)=NOT(IU(2))                          00357500
      IO(1)=NOT(IO(1))                          00357600
      IO(2)=NOT(IO(2))                          00357700
      ID(1)=NOT(ID(1))                          00357800
      ID(2)=NOT(ID(2))                          00357900
C
      1000 CONTINUE                             00358000
      READ ( 11 , '(A80)', END=2000 ) BUFF*W(20)(1:80) 00358100
      IF ( BUFF*W(20)(1:4) .EQ. D(1) .AND.      00358200
      +   BUFF*W(20)(5:5) .EQ. D(2)(1:1) ) THEN 00358300
      CALL ERROR ( 1014 )                       00358400
      GO TO 888                                  00358500
      END IF                                     00358600
      GO TO 1000                                00358700
C
      2000 CONTINUE                             00358800
      REWIND 11                                 00358900
C
      2100 CONTINUE                             00359000
      READ ( 11 , '(A80)', END=2200 ) BUFF*W(20)(1:80) 00359100
      WRITE(IPW , '(A80)') BUFF*W(20)(1:80)      00359200
      GO TO 2100                                00359300
C
      2200 CONTINUE                             00359400
      WRITE(IPW,'(6(A4))') D(1),D(2),U(1),U(2),O(1),O(2) 00359500
      CALL ERROR ( 1015 )                       00359600
C
      888 CONTINUE                              00359700
      REWIND IPW                                00359800
      CLOSE ( 11 )                             00359900
C==> CALL COMAND ( WORK(3) )                   00360000
      999 RETURN                                00360100
      END                                        00360200

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
      SUBROUTINE UDELET ( WORK )                00361100
C
C=====
C= FUNCTION =                                00361200
C= TO DELETE THE USER TO PASSWRD FILE =      00361300
C= PARAMETER =                                00361400
C= WORK : TSS COMAND AREA =                   00361500
C=====
C
C-----
C- PROCESS NOW CONDITION                      00361600
C-----
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC 00361700
C
C CHARACTER LOGID*8, PDATE*8, PASSWD*4        00361800
C INTEGER*2 IUS*OW, IMISS, IFUNC              00361900
C-----
C- DATABASE MANAGEMENT AREA                  00362000
C-----
COMMON / MANAG / LRECOB, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID      00362100
C
C CHARACTER ITNAME*24, ITFORM*24, SYSID*5    00362200
C INTEGER*2 LRECOB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE                    00362300
C-----
C- PROCESS CONDITION AREA                    00362400
C-----
COMMON / CONdit / JC(3)                      00362500
C
C INTEGER*2 JC                                00362600
C-----
C- ERROR MESSAGE AREA                        00362700
C-----
COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50) 00362800
C
C CHARACTER MFLAG*2, MSG*60                   00362900
C INTEGER*2 NMSG, NERCD                       00363000
C-----
C- UNIT INPUT OUTPUT NUMBER                  00363100
C-----
COMMON / UNIT / IOB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*, IPW               00363200
C
C INTEGER*2 IOB, IADDR, ILOG, IHSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*, IPW                                00363300
C-----
C- ADDR FILES DATA                          00363400
C-----

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
COMMON / ADDRWR / NDBAD(32000), NITAD(16,32000), 00366600
+ NMATCH(32000),NCON, NDOM, 00366700
+ ISEL(32000), IPRO(16), INUH(16), 00366800
+ IS(16), IE(16), NKEY 00366900
C
INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUH, IS, IE, NKEY 00367000
LOGICAL*1 ISEL, IPRO 00367100
C----- 00367200
C----- 00367300
C----- 00367400
C- DB FILE EMPTY RECORD CONTROL - 00367500
C----- 00367600
COMMON / BITMAP / MAP 00367700
C----- 00367800
CHARACTER MAP*4000 00367900
C----- 00368000
C----- 00368100
C----- 00368200
C- TERMTYPE COMMON AREA - 00368300
C----- 00368400
COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD 00368500
C----- 00368600
CHARACTER TNAME*8 00368700
C----- 00368800
C----- 00368900
C- FILE INPUT/OUTPUT BUFFER - 00369000
C----- 00369100
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL 00369200
C----- 00369300
CHARACTER BUFF*W*384, BUFF=1312, COND*384, CSEL*384 00369400
C----- 00369500
C----- 00369600
C- PICTURE COMMON AREA - 00369700
C----- 00369800
COMMON / PICT / NIA, PIA, LSIA(32) 00369900
C----- 00370000
INTEGER PIA, LSIA 00370100
CHARACTER NIA*16 00370200
C----- 00370300
CHARACTER U(2)*4,D(2)*4,D(2)*4, WORK(3)*50 00370400
INTEGER IU(2), IO(2), ID(2) 00370500
EQUIVALENCE (U,IU),(O,IO),(D,IO) 00370600
C----- 00370700
LOGICAL*1 FLG 00370800
C----- 00370900
C==> START 00371000
C----- 00371100
FLG = .FALSE. 00371200
C----- 00371300
IF ( BUFF(02:09) .EQ. 8H ) THEN 00371400
CALL ERROR ( 1012 ) 00371500
GO TO 999 00371600
END IF 00371700
C----- 00371800
CALL COMAND ( WORK(1) ) 00371900
CALL COMAND ( WORK(2) ) 00372000
OPEN ( 11 )

```

DATA SET NAME : 'Q402#.AEG.FORT77(BAHAMA)'

```

.....1.....2.....3.....4.....5.....6.....7.....8
C----- 00372100
D(1) = BUFF(02:05) 00372200
D(2) = BUFF(06:06) 00372300
C----- 00372400
ID(1)=NOT(ID(1)) 00372500
ID(2)=NOT(ID(2)) 00372600
C----- 00372700
1000 CONTINUE 00372800
READ ( 11, '(A80)', END=2000 ) BUFF*W(20)(1:80) 00372900
IF ( BUFF*W(20)(1:4) .EQ. D(1) .AND. 00373000
+ BUFF*W(20)(5:5) .EQ. D(2)(1:1) ) THEN 00373100
FLG = .TRUE. 00373200
GO TO 1000 00373300
END IF 00373400
WRITE( IPW, '(A80)' ) BUFF*W(20) 00373500
GO TO 1000 00373600
C----- 00373700
2000 CONTINUE 00373800
C----- 00373900
IF ( FLG ) THEN 00374000
CALL ERROR ( 1021 ) 00374100
ELSE 00374200
CALL ERROR ( 1018 ) 00374300
END IF 00374400
C----- 00374500
888 CONTINUE 00374600
REWIND IPW 00374700
CLOSE ( 11 ) 00374800
C----- 00374900
999 RETURN 00375000
END 00375100

```

DATE : 88/10/25


```

.....1.....2.....3.....4.....5.....6.....7.....8
SUBROUTINE ULIST ( WORK )
C
C-----
C= FUNCTION =
C= TO LIST UP THE USER FROM PASSOWRD FILE =
C= PARAMETER =
C= WORK : TSS COMAND AREA =
C-----
C
C- PROCCSS NOW CONDITION -
C-----
C COMMON / PROSS / LOGID, PDATE, IUSVOW, PASSWD, IMISS, IFUNC
C
C CHARACTER LOGID*8, PDATE*8, PASSWD*4
C INTEGER*2 IUSVOW, IMISS, IFUNC
C-----
C- DATABASE MANAGEMENT AREA -
C-----
C COMMON / MANAG / LRECD8, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
C CHARACTER ITNAME*24, ITFORM*24, SYSID*5
C INTEGER*2 LRECD8, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE
C-----
C- PROCESS CONDITION AREA -
C-----
C COMMON / CONDIT / JC(3)
C
C INTEGER*2 JC
C-----
C- ERROR MESSAGE AREA -
C-----
C COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)
C
C CHARACTER MFLAG*2, MSG*60
C INTEGER*2 NMSG, NERCD
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
C COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*, IPW
C
C INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*, IPW
C-----
C- ADDR FILES DATA -
C-----

```

```

00375200
00375300
00375400
00375500
00375600
00375700
00375800
00375900
00376000
00376100
00376200
00376300
00376400
00376500
00376600
00376700
00376800
00376900
00377000
00377100
00377200
00377300
00377400
00377500
00377600
00377700
00377800
00377900
00378000
00378100
00378200
00378300
00378400
00378500
00378600
00378700
00378800
00378900
00379000
00379100
00379200
00379300
00379400
00379500
00379600
00379700
00379800
00379900
00380000
00380100
00380200
00380300
00380400
00380500
00380600

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
COMMON / ADDR#R / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000), NCON, NDOH,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
C INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOH, INUM, IS, IE, NKEY
C LOGICAL*1 ISEL, IPRO
C-----
C- DB FILE EMPTY RECORD CONTROL -
C-----
C COMMON / BITHAP / MAP
C
C CHARACTER MAP*4000
C-----
C- TERMTYPE COMMON AREA -
C-----
C COMMON / TERM / IRO, IRI, TNAME, ITYPE, IRCD
C
C CHARACTER TNAME*8
C-----
C- FILE INPUT/OUTPUT BUFFER -
C-----
C COMMON / INOUT / BUFF#W(20), BUFF, COND, CSEL
C
C CHARACTER BUFF#W*384, BUFF*1312, COND*384, CSEL*384
C-----
C- PICTURE COMMON AREA -
C-----
C COMMON / PICT / MIA, PIA, LSIA(32)
C
C INTEGER PIA, LSIA
C CHARACTER MIA*16
C
C CHARACTER U(2)*4, D(2)*4, D(2)*4, WORK(5)*50
C INTEGER IU(2), ID(2), ID(2)
C EQUIVALENCE (U, IU), (D, ID), (D, ID)
C
C==> START
C
C CALL COMAND ( WORK(1) )
C CALL COMAND ( WORK(2) )
C OPEN ( 11 )
C
C 1000 CONTINUE
C READ ( IPW, '(A80)', END=2000 ) BUFF#W(20)(1:80)
C D(1) = BUFF#W(20)(01:04)
C D(2) = BUFF#W(20)(05:08)
C U(1) = BUFF#W(20)(09:12)
C U(2) = BUFF#W(20)(13:16)
C O(1) = BUFF#W(20)(17:20)
C O(2) = BUFF#W(20)(21:24)

```

```

00380700
00380800
00380900
00381000
00381100
00381200
00381300
00381400
00381500
00381600
00381700
00381800
00381900
00382000
00382100
00382200
00382300
00382400
00382500
00382600
00382700
00382800
00382900
00383000
00383100
00383200
00383300
00383400
00383500
00383600
00383700
00383800
00383900
00384000
00384100
00384200
00384300
00384400
00384500
00384600
00384700
00384800
00384900
00385000
00385100
00385200
00385300
00385400
00385500
00385600
00385700
00385800
00385900
00386000
00386100

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C
  IU(1)=NOT(IU(1))
  IU(2)=NOT(IU(2))
  IO(1)=NOT(IO(1))
  IO(2)=NOT(IO(2))
  ID(1)=NOT(ID(1))
  ID(2)=NOT(ID(2))
C
  WRITE ( 11 , '(6A4)' ) D(1),D(2),U(1),U(2),O(1),O(2)
  GO TO 1000
C
2000 CONTINUE
  REWIND IPW
  CLOSE ( 11 )
C
999 RETURN
  END
  
```

```

00386200
00386300
00386400
00386500
00386600
00386700
00386800
00386900
00387000
00387100
00387200
00387300
00387400
00387500
00387600
00387700
00387800
  
```

.....1.....2.....3.....4.....5.....6.....7.....8

```

C*****
C  SUBROUTINE UPDATE *
C*****
  SUBROUTINE UPDATE
C
C-----
C-  PROCCSS NOW CONDITION -
C-----
  COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC
C
  CHARACTER LOGID*8, PDATE*8, PASSWD*8
  INTEGER*2 IUS*OW, IMISS, IFUNC
C
C-----
C-  DATABASE MANAGEMENT AREA -
C-----
  COMMON / MANAG / LRECD*8, LRMAX, LRNOW, ITMAX, ITNOW,
  + ITNAME(16), ITFORM(16), ITCHAR(16),
  + ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
  CHARACTER ITNAME*24, ITFORM*24, SYSID*5
  INTEGER*2 LRECD*8, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
  + ITEMPE, IDATAS, IDATAE
C
C-----
C-  ADDR FILES DATA -
C-----
  COMMON / ADDR*8 / NDBAD(32000), NITAD(16,32000),
  + NMATCH(32000), NCON, NDOM,
  + ISEL(32000), IPRO(16), INUM(16),
  + IS(16), IE(16), NKEY
C
  INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
  LOGICAL*1 ISEL, IPRO
C
C-----
C-  PROCESS CONDITION AREA -
C-----
  COMMON / CONDIT / JC(3)
C
  INTEGER*2 JC
C
C-----
C-  FILE INPUT/OUTPUT BUFFER -
C-----
  COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
  CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C
C-----
C-  ERROR MESSAGE AREA -
C-----
  COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)
C
  CHARACTER MFLAG*2, HSG*60
  
```

```

00387900
00388000
00388100
00388200
00388300
00388400
00388500
00388600
00388700
00388800
00388900
00389000
00389100
00389200
00389300
00389400
00389500
00389600
00389700
00389800
00389900
00390000
00390100
00390200
00390300
00390400
00390500
00390600
00390700
00390800
00390900
00391000
00391100
00391200
00391300
00391400
00391500
00391600
00391700
00391800
00391900
00392000
00392100
00392200
00392300
00392400
00392500
00392600
00392700
00392800
00392900
00393000
00393100
00393200
00393300
  
```

- 117 -

```

.....1.....2.....3.....4.....5.....6.....7.....8
C      INTEGER*2 NMSG, NERCD      00393400
C-----
C-    RELATIVE ADDRESS AREA      00393500
C-----
C      COMMON / ISOU      / ISO    00393600
C      INTEGER*4 ISO          00393700
C-----
C-    UPDATE PROC USED DATA      00393800
C-----
C      CHARACTER PW*8 , UID*8 , UPW*8 , OPW*8
      INTEGER*4 IRAD(20)          00393900
C      = = UPDATE PROC          00394000
      IF( ISO .NE. 0 .AND.      00394100
      +   BUFF(225:249) .EQ. 25H      .AND.      00394200
      +   BUFF(250:274) .EQ. 25H      ) THEN      00394300
      DO 100 I = 0 , 360 , 24      00394400
          COND(I+1:I+24) = 24H      00394500
      100 CONTINUE                00394600
          COND(IS(1):IE(2)) = BUFF(11:74)      00394700
          COND(IS(3):IE(4)) = BUFF*W(1)(IS(3):IE(4))      00394800
          COND(IS(5):IE(5)) = LOGID(1:5)      00394900
          COND(IS(6):IE(6)) = PDATE      00395000
          COND(IS(7):IE(12)) = BUFF(75:224)      00395100
          ISEL(ISO) = .TRUE.      00395200
          CALL CHGDB ( IER )      00395300
          CALL ERROR ( 2301 )      00395400
          JC(1) = 2      00395500
          ISEL(ISO) = .FALSE.      00395600
          ISO = 0      00395700
          GOTO 999      00395800
      ENDIF      00395900
C      CALL SEARCH      00396000
C-----
C      IF( ISO .NE. 0 .AND. BUFF*W(19)(304:304) .NE. '0' ) THEN      00396100
      ISTART = ISO      00396200
      IOIM = +1      00396300
      CALL GETDB( ISTART , IOIM , IEND , ICO , IRAD , IER )      00396400
      IF( LOGID(1:5) .NE. BUFF*W(1)(65:69) .AND.      00396500
      +   LOGID(1:5) .NE. BUFF*W(1)(78:82) ) THEN      00396600
      WRITE(6,200) BUFF*W(1)(65:69),BUFF*W(1)(78:82)      00396700
      200 FORMAT(' PLEASE INPUT THE PASSWORD IN "',      00396800
      +   A5,' OR "',A5,'"')      00396900
      CALL PROMPT(2,' INPUT THE PASSWORD ==> ',IRC,0)      00397000
      READ (5,'(A8)') PW      00397100
      * = 0      00397200
      UID(1:5) = BUFF*W(1)(65:69)      00397300
      CALL BAHAPW ( UID , UPW , OPW )      00397400
      IF( PW .NE. UPW .AND. PW .NE. OPW ) * = * + 1      00397500
      IF( BUFF*W(1)(78:82) .NE. 5H      ) THEN      00397600
          UID(1:5) = BUFF*W(1)(78:82)      00397700
      ENDIF

```

DATA SET NAME : 'Q402#.AEG.FORT77(BAHAMA)'

```

.....1.....2.....3.....4.....5.....6.....7.....8
      CALL BAHAPW ( UID , UPW , OPW )      00398900
      IF( PW .NE. UPW .AND. PW .NE. OPW ) * = * + 1      00399000
      END IF      00399100
      IF( * .GE. 2 ) THEN      00399200
          CALL ERROR ( 2302 )      00399300
          BUFF*W(19)(304:304) = '3'      00399400
          JC(1) = 0      00399500
          GOTO 999      00399600
      ENDIF      00399700
      ENDIF      00399800
      BUFF(011:074) = BUFF*W(1)(IS(1):IE(2))      00399900
      BUFF(075:224) = BUFF*W(1)(IS(7):IE(12))      00400000
      BUFF*W(19)(304:304) = '0'      00400100
      ENDIF      00400200
999 RETURN      00400300
      END      00400400

```

DATE : 88/10/25

.....1.....2.....3.....4.....5.....6.....7.....8

```

SUBROUTINE USERCH ( WORK )
C
C=====
C= FUNCTION =
C= TO SEARCH THE USER FROM PASSWRD FILE =
C= PARAMETER =
C= WORK : TSS COMAND AREA =
C=====
C
C-----
C- PROCCSS NOW CONDITION
C-----
C
COMMON / PROSS / LOGID, PDATE, IUSWOW, PASSWD, IMISS, IFUNC
C
CHARACTER LOGID*8, PDATE*8, PASSWD*4
INTEGER*2 IUSWOW, IMISS, IFUNC
C-----
C- DATABASE MANAGEMENT AREA
C-----
C
COMMON / MAHAG / LRECD, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
CHARACTER ITNAME*24, ITFORM*24, SYSID*5
INTEGER*2 LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE
C-----
C- PROCESS CONDITION AREA
C-----
C
COMMON / CONdit / JC(3)
C
INTEGER*2 JC
C-----
C- ERROR MESSAGE AREA
C-----
C
COMMON / ERRORM / NHSG, MFLAG, NERCD(50), MSG(50)
C
CHARACTER MFLAG*2, MSG*60
INTEGER*2 NMSG, NERCD
C-----
C- UNIT INPUT OUTPUT NUMBER
C-----
C
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORK*, IPW
C
INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORK*, IPW
C-----
C- ADDR FILES DATA
C-----

```

```

00400500
00400600
00400700
00400800
00400900
00401000
00401100
00401200
00401300
00401400
00401500
00401600
00401700
00401800
00401900
00402000
00402100
00402200
00402300
00402400
00402500
00402600
00402700
00402800
00402900
00403000
00403100
00403200
00403300
00403400
00403500
00403600
00403700
00403800
00403900
00404000
00404100
00404200
00404300
00404400
00404500
00404600
00404700
00404800
00404900
00405000
00405100
00405200
00405300
00405400
00405500
00405600
00405700
00405800
00405900

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

COMMON / ADDRVR / NDBAD(32000), NITAD(16,32000),
+ NNATCH(32000), NCON, NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
INTEGER*2 NDBAD, NITAD, NNATCH, NCON, NDOM, INUM, IS, IE, NKEY
LOGICAL*1 ISEL, IPRO
C-----
C- DB FILE EMPTY RECORD CONTROL
C-----
C
COMMON / BITMAP / MAP
C
CHARACTER MAP*4000
C-----
C- TERMTYPE COMMON AREA
C-----
C
COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD
C
CHARACTER TNAME*8
C-----
C- FILE INPUT/OUTPUT BUFFER
C-----
C
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C-----
C- PICTURE COMMON AREA
C-----
C
COMMON / PICT / MIA, PIA, LSIA(32)
C
INTEGER PIA, LSIA
CHARACTER MIA*16
C
CHARACTER U(2)*4, O(2)*4, D(2)*4, WORK(3)*50
INTEGER IU(2), IO(2), ID(2)
EQUIVALENCE (U,IU),(O,IO),(D,ID)
C=> START
C
IF ( BUFF(02:09) .EQ. 8H ) THEN
CALL ERROR ( 1012 )
GO TO 888
END IF
C
CALL COMAND ( WORK(3) )
OPEN ( 11 )
C
D(1) = BUFF(02:05)
D(2) = BUFF(06:06)
C
ID(1)=NOT(ID(1))

```

```

00406000
00406100
00406200
00406300
00406400
00406500
00406600
00406700
00406800
00406900
00407000
00407100
00407200
00407300
00407400
00407500
00407600
00407700
00407800
00407900
00408000
00408100
00408200
00408300
00408400
00408500
00408600
00408700
00408800
00408900
00409000
00409100
00409200
00409300
00409400
00409500
00409600
00409700
00409800
00409900
00410000
00410100
00410200
00410300
00410400
00410500
00410600
00410700
00410800
00410900
00411000
00411100
00411200
00411300
00411400

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
      ID(2)=NOT(ID(2))                                00411500
C      1000 CONTINUE                                  00411600
      READ ( 11 , '(A80)' , END=2000 ) BUFF*W(20)(1:80) 00411700
C      IF ( BUFF*W(20)(1:4) .EQ. D(1) .AND.           00411800
+      BUFF*W(20)(5:5) .EQ. D(2)(1:1) ) THEN        00411900
      U(1) = BUFF*W(20)(09:12)                        00412000
      U(2) = BUFF*W(20)(13:16)                        00412100
      O(1) = BUFF*W(20)(17:20)                        00412200
      O(2) = BUFF*W(20)(21:24)                        00412300
      IU(1)=NOT(IU(1))                                00412400
      IU(2)=NOT(IU(2))                                00412500
      IO(1)=NOT(IO(1))                                00412600
      IO(2)=NOT(IO(2))                                00412700
      BUFF(10:13)=U(1)                                 00412800
      BUFF(14:17)=U(2)                                 00412900
      BUFF(18:21)=O(1)                                 00413000
      BUFF(22:25)=O(2)                                 00413100
      GO TO 888                                        00413200
      END IF                                           00413300
C      GO TO 1000                                     00413400
C      2000 CONTINUE                                  00413500
      CALL ERROR ( 1018 )                              00413600
C      888 CONTINUE                                   00413700
      CLOSE ( 11 )                                     00413800
C      RETURN                                         00413900
      END                                              00414000

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
      SUBROUTINE USMENT                                00414700
C      FUNCTION                                         00414800
C=      TO DISPLAY THE USER MENTE. PICTURE           00414900
C=      =                                             00415000
C-----                                             00415100
C-      PROCESS NOW CONDITION                         00415200
C-----                                             00415300
COMMON / PROSS / LOGID, PDATE, IUS*OW, PASSWD, IMISS, IFUNC 00415400
C      CHARACTER LOGID*8, PDATE*8, PASSWD*4           00415500
      INTEGER*2 IUS*OW, IMISS, IFUNC                 00415600
C-----                                             00415700
C-      DATABASE MANAGEMENT AREA                     00415800
C-----                                             00415900
COMMON / MANAG / LRECCDB, LRMAX, LRNOW, ITMAX, ITNOW,      00416000
+      ITNAME(16), ITFORM(16), ITCHAR(16),            00416100
+      ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID          00416200
C      CHARACTER ITNAME*24, ITFORM*24, SYSID*5       00416300
      INTEGER*2 LRECCDB, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS, 00416400
+      ITEMPE, IDATAS, IDATAE                        00416500
C-----                                             00416600
C-      PROCESS CONDITION AREA                       00416700
C-----                                             00416800
COMMON / CONDOT / JC(3)                                00416900
C      INTEGER*2 JC                                   00417000
C-----                                             00417100
C-      ERROR MESSAGE AREA                           00417200
C-----                                             00417300
COMMON / ERRORM / NMSG, MFLAG, NERCD(50), MSG(50)        00417400
C      CHARACTER MFLAG*2, MSG*60                     00417500
      INTEGER*2 NMSG, NERCD                          00417600
C-----                                             00417700
C-      UNIT INPUT OUTPUT NUMBER                     00417800
C-----                                             00417900
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,    00418000
+      ITSSIN, ITSSOT, IWORK*, IPW                   00418100
C      INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT, 00418200
+      IWORK*, IPW                                    00418300
C-----                                             00418400
C-      ADDR FILES DATA                             00418500
C-----                                             00418600
COMMON / ADDR*R / NDBAD(32000), NITAD(16,32000),        00418700
+      NHATCH(32000), NCON, NOOM,                    00418800

```

- 120 -

.....1.....2.....3.....4.....5.....6.....7.....8

```

+          ISEL(32000), IPRO(16), INUM(16),      00420200
+          IS(16), IE(16), NKEY                    00420300
C
C   INTEGER*2 HDBAD, HITAD, NMATCH, NCON, NDM, INUM, IS, IE, NKEY  00420400
C   LOGICAL*1 ISEL, IPRO                                00420500
C-----
C-  DB FILE EMPTY RECORD CONTROL                      00420600
C-----
C   COMMON / BITHAP / HAP                             00420700
C   CHARACTER MAP*4000                                00420800
C-----
C-  TERMTYPE COMMON AREA                              00420900
C-----
C   COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD     00421000
C   CHARACTER TNAME*8                                 00421100
C-----
C-  FILE INPUT/OUTPUT BUFFER                          00421200
C-----
C   COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL    00421300
C   CHARACTER BUFF*W*384, BUFF=1312, COND=384, CSEL*384 00421400
C-----
C-  PICTURE COMMON AREA                              00421500
C-----
C   COMMON / PICT / MIA, PIA, LSIA(32)                00421600
C   INTEGER PIA, LSIA                                  00421700
C   CHARACTER MIA*16                                  00421800
C   CHARACTER WK(3)*50                                00421900
C
C==> START                                           00422000
C
C   WK(1)='COPY '//SYSID//'-BAHAMA.PW'//1H'/' ' B'// 00422100
C   + PDATE(1:2)//PDATE(4:5)//PDATE(7:8) //' .WORK ' 00422200
C   WK(2)='ALLOC F(FT11F001) DA(0)//PDATE(1:2)//PDATE(4:5)// 00422300
C   + PDATE(7:8) //' .WORK ' SHR                      00422400
C   WK(3)='ALLOC F(FT11F001) ' //                      00422500
C   + 'DA(//1H'//SYSID//'-BAHAMA.PW'//1H'/' ' ) SHR 00422600
C
C   MIA = 'SYSTEMU USER '                              00422700
C   IF ( ITYPE .NE. 1 ) MIA = 'SYSTEMU USER* '        00422800
C
C 200 CONTINUE                                       00422900
C
C   IF ( IFG .EQ. 1 .OR. IFG .EQ. 0 ) THEN            00423000
C   DO 300 I=1,1251,50                                00423100
C 300 BUFF(I:I+49)=50H                                00423200
C   BUFF(1301:1312)=12H                               00423300
C   00423400
C   00423500
C   00423600
C   00423700
C   00423800
C   00423900
C   00424000
C   00424100
C   00424200
C   00424300
C   00424400
C   00424500
C   00424600
C   00424700
C   00424800
C   00424900
C   00425000
C   00425100
C   00425200
C   00425300
C   00425400
C   00425500
C   00425600

```

.....1.....2.....3.....4.....5.....6.....7.....8

```

END IF
IFG = 0
C
CDS
* WRITE(6,*) BUFF(1:30)
CDE
CALL IPFMIO ( LSIA , MIA , BUFF , IRCA )
C
IF ( IRCA .GT. 24 ) THEN
CALL ERROR ( 1011 )
GO TO 999
END IF
C
IF ( IRCA .EQ. 3 ) GO TO 999
IF ( BUFF(1:1) .EQ. ' ' ) THEN
CALL ERROR ( 1002 )
GO TO 200
END IF
C
IF ( BUFF(1:1) .NE. 1H1 .AND.
+ BUFF(1:1) .NE. 1H2 .AND.
+ BUFF(1:1) .NE. 1H3 .AND.
+ BUFF(1:1) .NE. 1H4 ) GO TO 200
C
READ (BUFF(1:1) , '(11)' ) IFG
C
IF ( IFG .EQ. 1 ) CALL UADD ( WK )
IF ( IFG .EQ. 2 ) CALL UUPDAT ( WK )
IF ( IFG .EQ. 3 ) CALL USERCH ( WK )
IF ( IFG .EQ. 4 ) CALL UDELET ( WK )
C
GO TO 200
C
999 CONTINUE
RETURN
END
00425700
00425800
00425900
00426000
00426100
00426200
00426300
00426400
00426500
00426600
00426700
00426800
00426900
00427000
00427100
00427200
00427300
00427400
00427500
00427600
00427700
00427800
00427900
00428000
00428100
00428200
00428300
00428400
00428500
00428600
00428700
00428800
00428900
00429000
00429100
00429200

```

```

+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8
SUBROUTINE UUPDAT ( WORK )
C
C=====
C= FUNCTION =
C= TO UPDATE THE USER TO PASSWRD FILE =
C= PARAMETER =
C= WORK : TSS COMAND AREA =
C=====
C
C-----
C- PROCCSS NOW CONDITION -
C-----
COMMON / PROSS / LOGID, PDATE, IUSYOW, PASSWD, IMISS, IFUNC
C
C CHARACTER LOGID*8, PDATE*8, PASSWD*4
C INTEGER*2 IUSYOW, IMISS, IFUNC
C-----
C- DATABASE MANAGEMENT AREA -
C-----
COMMON / MANAG / LRECD, LRMAX, LRNOW, ITMAX, ITNOW,
+ ITNAME(16), ITFORM(16), ITCHAR(16),
+ ITEMPS, ITEMPE, IDATAS, IDATAE, SYSID
C
C CHARACTER ITNAME*24, ITFORM*24, SYSID*5
C INTEGER*2 LRECD, LRMAX, LRNOW, ITMAX, ITNOW, ITCHAR, ITEMPS,
+ ITEMPE, IDATAS, IDATAE
C-----
C- PROCESS CONDITION AREA -
C-----
COMMON / CONDIT / JC(3)
C
C INTEGER*2 JC
C-----
C- ERROR MESSAGE AREA -
C-----
COMMON / ERRORM / NMSG, MFLAG, MERCD(50), MSG(50)
C
C CHARACTER MFLAG*2, HSG*60
C INTEGER*2 NMSG, MERCD
C-----
C- UNIT INPUT OUTPUT NUMBER -
C-----
COMMON / UNIT / IDB, IADDR, ILOG, IMSG, IWORK, ILIST,
+ ITSSIN, ITSSOT, IWORKY, IPW
C
C INTEGER*2 IDB, IADDR, ILOG, IMSG, IWORK, ILIST, ITSSIN, ITSSOT,
+ IWORKY, IPW
C-----
C- ADDR FILES DATA -
C-----

```

```

00429300
00429400
00429500
00429600
00429700
00429800
00429900
00430000
00430100
00430200
00430300
00430400
00430500
00430600
00430700
00430800
00430900
00431000
00431100
00431200
00431300
00431400
00431500
00431600
00431700
00431800
00431900
00432000
00432100
00432200
00432300
00432400
00432500
00432600
00432700
00432800
00432900
00433000
00433100
00433200
00433300
00433400
00433500
00433600
00433700
00433800
00433900
00434000
00434100
00434200
00434300
00434400
00434500
00434600
00434700

```

```

+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8
COMMON / ADDR*8 / NDBAD(32000), NITAD(16,32000),
+ NMATCH(32000),NCON,NDOM,
+ ISEL(32000), IPRO(16), INUM(16),
+ IS(16), IE(16), NKEY
C
C INTEGER*2 NDBAD, NITAD, NMATCH, NCON, NDOM, INUM, IS, IE, NKEY
C LOGICAL*1 ISEL, IPRO
C-----
C- DB FILE EMPTY RECORD CONTROL -
C-----
COMMON / BITMAP / MAP
C
C CHARACTER MAP*4000
C-----
C- TERMTYPE COMMON AREA -
C-----
COMMON / TERM / IRO, IR1, TNAME, ITYPE, IRCD
C
C CHARACTER TNAME*8
C-----
C- FILE INPUT/OUTPUT BUFFER -
C-----
COMMON / INOUT / BUFF*W(20), BUFF, COND, CSEL
C
C CHARACTER BUFF*W*384, BUFF*1312, COND*384, CSEL*384
C-----
C- PICTURE COMMON AREA -
C-----
COMMON / PICT / MIA, PIA, LSIA(32)
C
C INTEGER PIA, LSIA
C CHARACTER MIA*16
C
C CHARACTER U(2)*4,D(2)*4,D(2)*4, WORK(3)*50
C INTEGER IU(2),IO(2),ID(2)
C EQUIVALENCE (U,IU),(D,IO),(D,ID)
C
C LOGICAL*1 FLG
C
C=> START
C
C FLG = .FALSE.
C
C IF ( BUFF(02:09) .EQ. 8H ) THEN
C CALL ERROR ( 1012 )
C GO TO 999
C END IF
C
C IF ( BUFF(10:17) .EQ. 8H .AND.
+ BUFF(18:25) .EQ. 8H ) THEN
C CALL ERROR ( 1013 )

```

```

00434800
00434900
00435000
00435100
00435200
00435300
00435400
00435500
00435600
00435700
00435800
00435900
00436000
00436100
00436200
00436300
00436400
00436500
00436600
00436700
00436800
00436900
00437000
00437100
00437200
00437300
00437400
00437500
00437600
00437700
00437800
00437900
00438000
00438100
00438200
00438300
00438400
00438500
00438600
00438700
00438800
00438900
00439000
00439100
00439200
00439300
00439400
00439500
00439600
00439700
00439800
00439900
00440000
00440100
00440200

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
      GO TO 999                                00440300
      END IF                                    00440400
C
      IF ( BUFF(10:17) .EQ. BUFF(18:25) ) THEN 00440500
      CALL ERROR ( 1017 )                      00440600
      GO TO 999                                00440700
      END IF                                    00440800
C
      CALL COMAND ( WORK(1) )                  00440900
      CALL COMAND ( WORK(2) )                  00441000
      OPEN ( 11 )                              00441100
C
      D(1) = BUFF(02:05)                       00441200
      D(2) = BUFF(06:06)/// ' '                00441300
      U(1) = BUFF(10:13)                       00441400
      U(2) = BUFF(14:17)                       00441500
      O(1) = BUFF(18:21)                       00441600
      O(2) = BUFF(22:25)                       00441700
C
      IU(1)=NOT(IU(1))                          00441800
      IU(2)=NOT(IU(2))                          00441900
      IO(1)=NOT(IO(1))                          00442000
      IO(2)=NOT(IO(2))                          00442100
      ID(1)=NOT(ID(1))                          00442200
      ID(2)=NOT(ID(2))                          00442300
C
1000 CONTINUE                                00442400
      READ ( 11 , '(A80)' , END=2000 ) BUFF*W(20)(1:80) 00442500
      IF ( .NOT.FLG .AND.                      00442600
      +   BUFF*W(20)(1:4) .EQ. D(1) .AND.      00442700
      +   BUFF*W(20)(5:5) .EQ. D(2)(1:1) ) THEN 00442800
      FLG = .TRUE.                              00442900
      BUFF*W(20)(09:12) = U(1)                  00443000
      BUFF*W(20)(13:16) = U(2)                  00443100
      BUFF*W(20)(17:20) = O(1)                  00443200
      BUFF*W(20)(21:24) = O(2)                  00443300
      WRITE( IPW , '(A80)' ) BUFF*W(20)         00443400
      GO TO 1000                                00443500
      END IF                                    00443600
      WRITE( IPW , '(A80)' ) BUFF*W(20)         00443700
      GO TO 1000                                00443800
C
2000 CONTINUE                                00443900
C
      IF ( FLG ) THEN                           00444000
      CALL ERROR ( 1020 )                       00444100
      ELSE                                       00444200
      CALL ERROR ( 1018 )                       00444300
      END IF                                    00444400
C
888 CONTINUE                                00444500
      REWIND IPW                                00444600
      CLOSE ( 11 )                              00444700
C
999 RETURN                                    00444800

```

```

.....1.....2.....3.....4.....5.....6.....7.....8
      END                                       00444900
                                           00445000
                                           00445100
                                           00445200
                                           00445300
                                           00445400
                                           00445500
                                           00445600
                                           00445700

```