

A.FICARRA, R.PRIMAVERA, M.TUGNOLI

**Procedure automatiche
per la registrazione delle presenze
e assenze del personale**

IRA 245/97

INDICE

INTRODUZIONE	1
PREREQUISITI PER L'UTILIZZO DELLE PROCEDURE	3
IL DATABASE	7
1. Tabella "CodiciAssenze"	8
2. Tabella "Dipendenti"	9
3. Tabella "Festivi"	13
4. Tabella "Movimenti"	14
5. Tabella "Provvisori"	14
6. Tabella "FerieInfo"	14
CARATTERISTICHE GENERALI DEI PROGRAMMI	15
PROGRAMMA "ANAPER"	19
1. Considerazioni generali	19
2. Avvio del programma	20
3. Descrizione della finestra di schermo	20
4. Selezione delle caselle e spostamenti	22
5. Aggiornamenti e controlli di validità	24
6. Inserimento di un nuovo record	26
7. Come rendere veloci le operazioni di inserimento sequenziale	26
8. Cancellazione di un record	27
9. Fine del programma e creazione del file di stampa	28
PROGRAMMA "ADSENZE"	29
1. Considerazioni generali	29
2. Avvio del programma	29
3. Attivazione di un "Prospetto"	30
4. Inserimento di nuove colonne nel Prospetto	33
5. Disattivazione di un Prospetto	35
6. Inserimento di "previsioni" di assenza	36
7. Visualizzazione e stampa delle tabelle di riepilogo	38
8. Rapporti mensili	41
9. Gestione della tabella "CodiciAssenze"	42
10. Comando "Frame"	43

PROGRAMMA “DIPCHECK”	45
1. Considerazioni generali	45
2. Avvio del programma	45
3. Selezione del nome di un dipendente	46
4. Selezione della “Situazione generale delle ferie”	49
5. Per gli utenti più “pigri”	51
PROGRAMMA “TRANSFER”	53
CONCLUSIONI E ULTIME RACCOMANDAZIONI	55

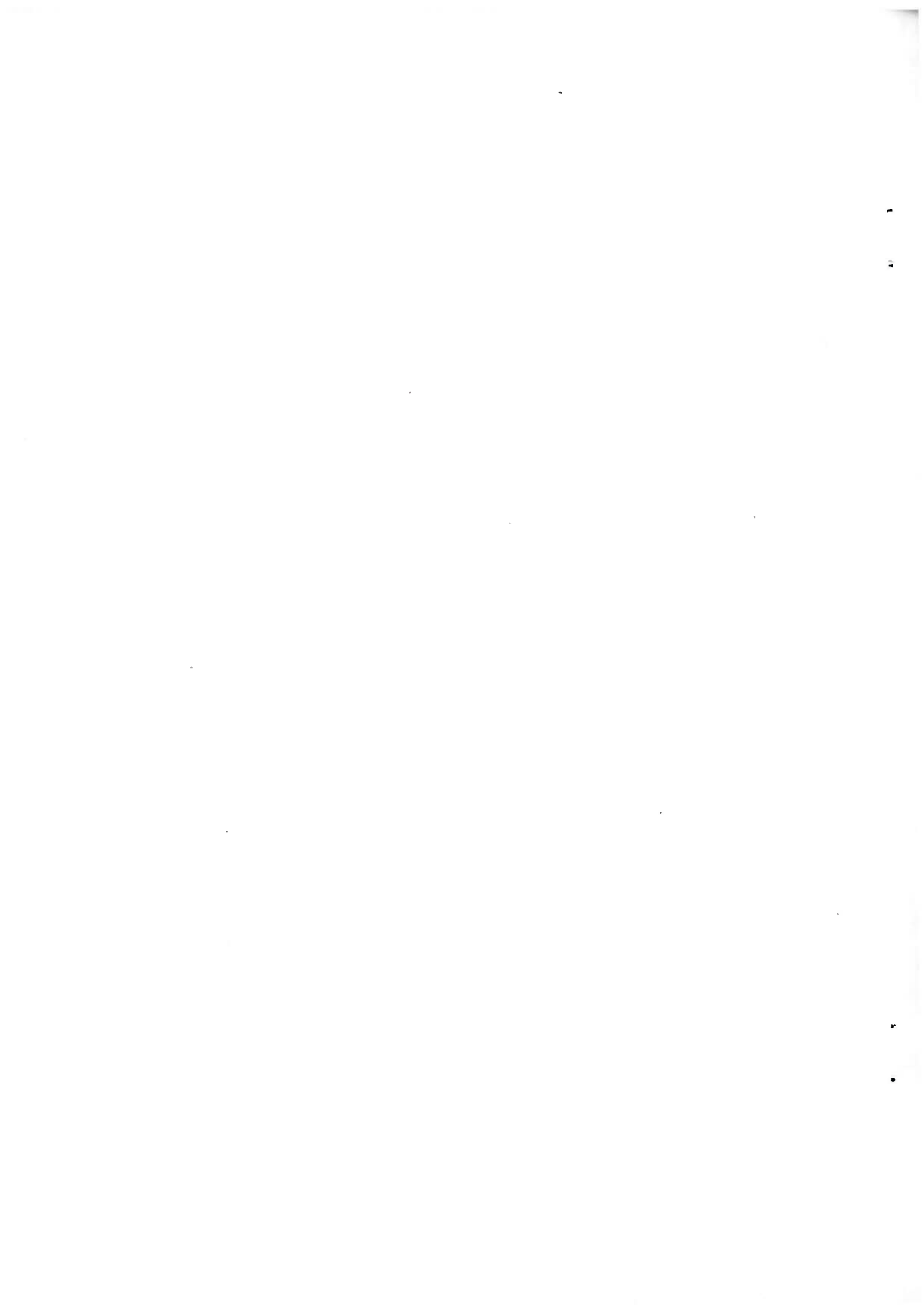
Introduzione

In attesa dell'installazione, nell'Area della Ricerca del CNR di Bologna, di un sistema automatico di controllo delle presenze e degli accessi (tramite rilevatori magnetici e relativo software di gestione), l'elenco delle presenze e assenze giornaliere dei dipendenti viene registrato, in ogni Istituto, su materiale cartaceo, che le segreterie si incaricano di raccogliere ed esaminare per ricavarne le informazioni sulla situazione di ogni singolo dipendente; tali informazioni, oltre a essere necessarie a livello locale (per esempio, per sapere quanti giorni di ferie ogni dipendente può ancora fare nell'anno in corso), devono essere inviate mensilmente alla sede centrale, in un apposito modulo, con l'indicazione dei giorni complessivi di presenza e delle date, e relative motivazioni, dei giorni di assenza.

Questo lavoro di aggregazione di dati giornalieri e di risuddivisione degli stessi su ogni singolo dipendente, è eseguito generalmente in modo manuale, e quindi è molto lungo, impegnativo e noioso. Nell'Istituto di Radioastronomia la persona che se ne occupava è andata di recente in pensione; ciò ha aperto drammaticamente un problema (come accade in tante altre circostanze analoghe!) e ha fornito l'occasione per cercare di rendere il lavoro più rapido e "piacevole", mediante introduzione di procedure automatiche. Va precisato subito che tali procedure sono da considerarsi provvisorie, e sostitutive dell'attuale lavoro manuale, ma non del futuro sistema di rilevazione automatica; infatti le procedure non prevedono che i dati vengano letti direttamente da dispositivi magnetici, ma che siano raccolti ancora su materiale cartaceo, e giornalmente vengano digitati e trasferiti nel calcolatore: solo da questo punto in poi si innesta l'elaborazione automatica (che copre in ogni caso la parte più rilevante del lavoro). Inoltre queste procedure non costituiscono un sistema completo di gestione del personale, in quanto, per esempio, non trattano né di "orari" né di "turni", ma solo di "giornate" intere; non esauriscono quindi tutti gli aspetti del problema, ma solo alcuni (per esempio il conteggio e il controllo delle ferie di ogni dipendente).

Le procedure consistono nel trattamento di dati memorizzati in un *database*, e utilizzano quattro programmi, scritti in linguaggio *Visual C++*. Questo Rapporto intende descrivere l'organizzazione dei dati e l'uso delle procedure, senza entrare nei particolari del linguaggio di programmazione; è pertanto rivolto a possibili utilizzatori e non a programmatori o informatici (questi, se sono interessati, possono consultare il **Rapporto IRA 239/97**: "Soluzioni Visual C++ adottate nei programmi di gestione del personale"). I quattro programmi sono:

- 1) **Anaper**: gestisce l'anagrafe del personale;
- 2) **Adsenze** (dal latino *adsum* = "sono presente"): gestisce le presenze e le assenze dei dipendenti, per ogni giornata lavorativa
- 3) **Dipcheck**: ricava informazioni individuali e statistiche sulle presenze e assenze dei dipendenti
- 4) **Transfer**: serve per riallineare le tabelle del *database* (da usarsi in casi eccezionali)



Prerequisiti per l'utilizzo delle procedure

I programmi per la gestione del personale, presentati in questo Rapporto, devono essere eseguiti da PC (*Personal Computer*), sotto sistema operativo **Windows 95** (o **Windows NT**).

Le informazioni risiedono tutte in un unico *database*, di tipo **MDB** (**Microsoft DataBase**), gestito dal sistema di accesso **ODBC** (**Open DataBase Connectivity**).

Programmi e dati possono anche risiedere in un PC autonomo (cioè non collegato alla rete), che, in questo caso, utilizzerebbe le procedure in modo esclusivo. Tuttavia si consiglia vivamente di memorizzare i files dei quattro programmi e del *database* in un *server* di rete, operante con **Windows NT**; questa soluzione è preferibile per i seguenti motivi:

- 1) l'utente può lanciare i programmi da qualunque PC collegato alla rete, purché provvisto di sistema operativo **Windows 95**;
- 2) l'uso dei programmi che modificano il *database* (tutti, salvo **Dipcheck**) deve essere ristretto al solo personale autorizzato: ciò è possibile, in quanto il *server* di rete può selezionare, tramite controllo della *password*, gli *Usernames* con diritto di accesso al *database* in modalità di "lettura-scrittura";
- 3) il programma **Dipcheck**, che invece opera in modalità di sola lettura, e che è stato fatto proprio allo scopo di permettere a tutti il libero accesso all'informazione, può essere eseguito senza restrizioni da qualunque punto della rete: ogni dipendente può, dal computer della sua scrivania, utilizzare **Dipcheck**, per esempio, per sapere quanti giorni di ferie gli restano da fare, oppure per verificare che le date delle sue assenze corrispondono a quanto gli risulta ecc...

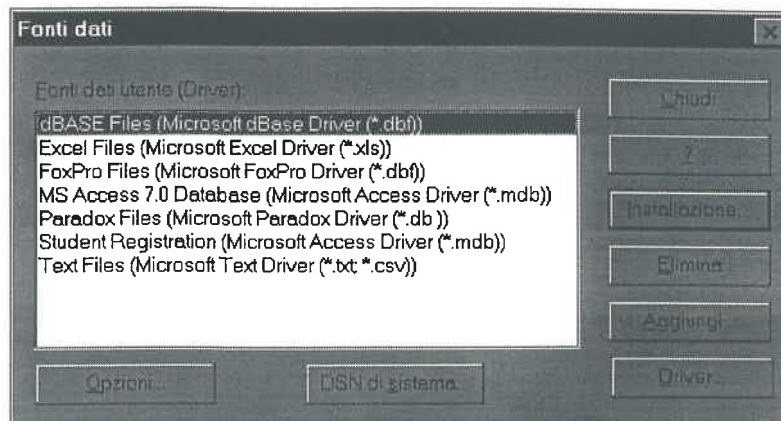
Qualunque sia il tipo di organizzazione (PC autonomo o in rete), tutti i PC da cui vengono lanciati i programmi devono essere provvisti delle sue seguenti componenti software:

- *driver* **ODBC** (normalmente inserito durante la stessa installazione di **Windows 95**)
- **Microsoft Access**

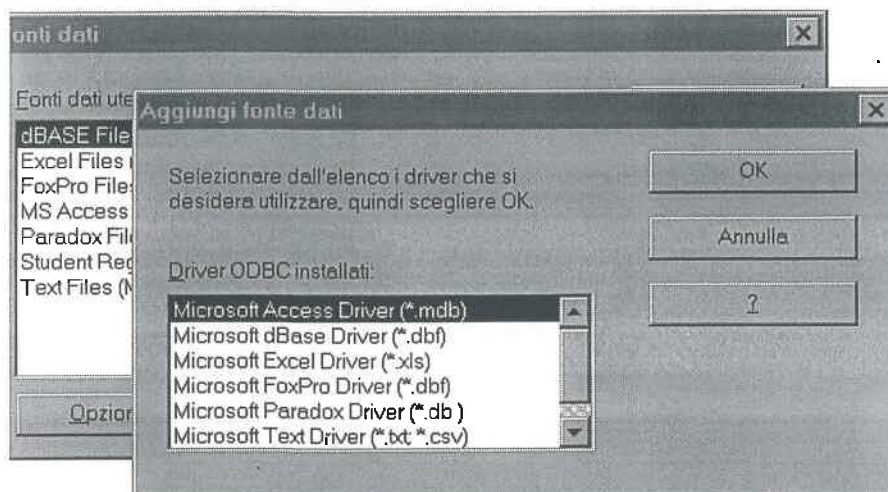
Entrambe le componenti servono per gestire il *database*: la prima permette ai programmi di riconoscere il file dei dati come *database* **ODBC**, la seconda, che è una ben nota applicazione del gruppo "**Office**", serve quando bisogna modificare il *database* direttamente, cioè senza utilizzare i programmi.

Inoltre, in ogni PC, prima di lanciare uno qualunque dei quattro programmi, è necessario eseguire *una tantum* la "registrazione" del *database*, associando al nome del file, che può essere scelto liberamente (il *database* dell'Istituto di Radioastronomia si chiama **DipIRA.mdb**), il nome del cosiddetto "**ODBC data source**", che invece è prefissato e rappresenta la stringa di identificazione del *database* da parte dei programmi; questa stringa (che ovviamente è la stessa per tutti e quattro i programmi) è "**Gestione Personale**" (notare la flessibilità di questo tipo di organizzazione, che permette di cambiare *database* senza modificare i programmi, ma semplicemente scegliendo un diverso file da associare all'**ODBC data source**).

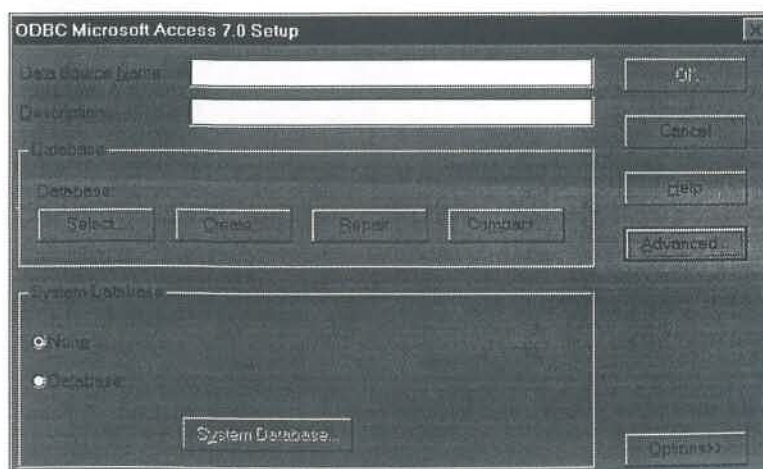
L'operazione si esegue lanciando dal *pannello di controllo* l'applicazione identificata dall'*icona ODBC 32 bit*, che fa apparire una finestra come nella seguente figura:



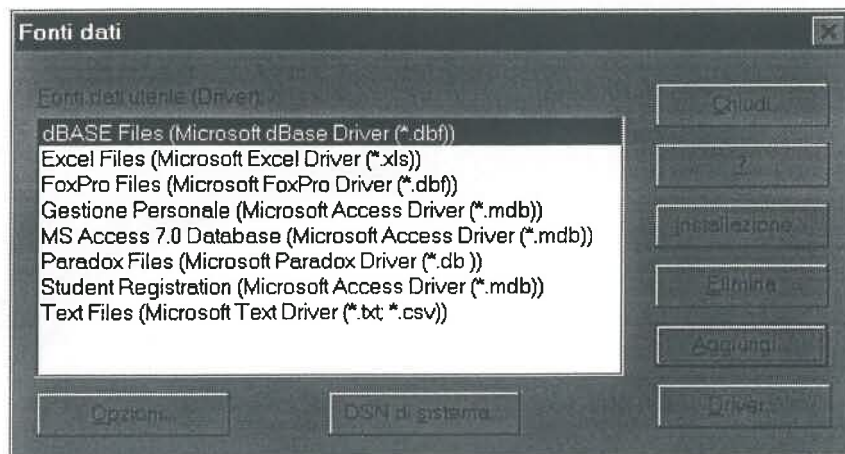
Selezionando la casella "Aggiungi ...", appare una seconda finestra:



A questo punto bisogna selezionare "**Microsoft Access Driver(*.mdb)**" (se non é già selezionato) e poi confermare con OK. Nella successiva finestra:



inserire la frase: **Gestione Personale** (con esatte maiuscole e minuscole e un solo spazio fra le due parole) nella casella “**Data Source name:**” e poi premere “Select ...” per trovare e selezionare (nel modo consueto di *Windows*) il nome del file fisico da associare al nome del **data source**. Confermata la selezione con OK, riappare la prima finestra:



in cui, come si può notare, il **data source** “**Gestione Personale**” é stato incluso nell’elenco (per eventuali modifiche o correzione di errori bisogna selezionare la voce che interessa e poi premere “Installazione ...”). Infine, premere “Chiudi” per terminare.

Gli Istituti interessati ad utilizzare le procedure possono, se dispongono di PC collegati alla rete locale dell’Area, trovare i programmi in formato eseguibile (**Anaper.exe** ecc...) nella “cartella” **EXPORT** del computer **HAVEN** del dominio **IRA**. Nello stesso posto é disponibile un “prototipo” di *database*, di nome “**Proto.mdb**”, che é già strutturato come necessario e contiene alcuni records indispensabili alla partenza. Il futuro utente dovrà copiare questo file e quelli dei programmi nel proprio PC (o, ancora meglio, in un *server* di rete) e poi cambiare nome al *database* (scegliendo un nome di fantasia) e utilizzare tale nome per la registrazione del **data source** “**Gestione Personale**” in tutti i PC da cui intende lanciare i programmi; successivamente dovrà definire sul *server* (se ha scelto questa soluzione) l’elenco degli *Usernames* autorizzati ad accedere al *database* in modalità lettura-scrittura (tutti gli altri saranno autorizzati ad accedere in sola lettura per eseguire **Dipcheck**).

Un’ultima raccomandazione: i programmi fanno largo uso delle date, memorizzate in varie parti del *database* come informazioni fondamentali, e spesso confrontano tali informazioni con la data del giorno in cui sta lavorando. E’ perciò indispensabile che la data di sistema di ogni PC sia assolutamente esatta (é importante che sia approssimativamente corretta anche l’ora, per evitare cambi di data durante il giorno!): provvedere a che questo requisito sia soddisfatto prima di installare le procedure.



Il Database

La conoscenza approfondita della struttura del *database* non è condizione indispensabile per l'utilizzo delle procedure, in quanto, in generale, i dati sono gestiti attraverso i programmi. Tuttavia è necessario che almeno una persona per Istituto possieda tale requisito, principalmente per due motivi:

- 1) come vedremo, una tabella del *database* è gestita non dai programmi ma tramite **Microsoft Access**;
- 2) in questo Rapporto, nella descrizione dei programmi si fa spesso riferimento alle "tabelle" del *database*.

In linea di principio, esisterebbe anche un terzo motivo: la necessità di intervenire direttamente sui dati, tramite **Access**, nell'eventualità di eccezionali malfunzioni o errori: in realtà, nello sviluppo dei programmi molta cura è stata spesa per evitare che errori operativi possano "rovinare" i dati (anzi, l'unica possibilità di rovinarli è proprio quella di accedervi direttamente con **Access** !); quanto a possibili "buchi" degli stessi programmi, confidiamo (molto modestamente) che non ve ne siano e, anche se ve ne fossero, servirebbe a poco tentare ogni volta di porre rimedio correggendo i *records* manualmente. Pertanto sconsigliamo vivamente di utilizzare **Access** per accedere a quelle tabelle del *database* la cui gestione è affidata ai programmi (cioè tutte salvo una), a parte casi eccezionalissimi e imprevedibili.

Il *database* (che, ricordiamo, è costituito da un unico file, con nome qualsiasi e estensione **.mdb**, registrato con **ODBC data source name "Gestione Personale"**), si compone di 6 "tabelle":

- 1) **CodiciAssenze**, utilizzata in lettura e scrittura da **Adsenze** e in sola lettura da **Dipcheck**;
- 2) **Dipendenti**, utilizzata in lettura e scrittura da **Anaper** e in sola lettura da **Adsenze** e **Dipcheck**;
- 3) **Festivi**, utilizzata in sola lettura da **Adsenze** (per le operazioni di gestione e aggiornamento bisogna utilizzare **Access**);
- 4) **Movimenti**, utilizzata in lettura e scrittura da **Adsenze** e in sola lettura da **Dipcheck**;
- 5) **Provvisori**, utilizzata in lettura e scrittura da **Adsenze** e in sola lettura da **Dipcheck**.
- 6) **FerieInfo**, utilizzata in lettura e scrittura da **Anaper** e **Adsenze** e in sola lettura da **Dipcheck**;

Procediamo ora con la descrizione di ogni singola tabella, precisando che il significato di alcuni suoi *campi* sarà meglio chiarito quando parleremo dell'utilizzo dei programmi.

1 - Tabella "CodiciAssenze"

Nome campo	Tipo dati	Descrizione
CodAss	Contatore	Codice presenza/assenza progressivo
Descr	Testo	Descrizione tipo di assenza
DescrHy	Testo	Descrizione con sillabazione
AssCode	Testo	Codice presenza/assenza legale
Preced	Numerico	Precedenza nella visualizzazione

La tabella contiene un *record* per ogni tipo di assenza, secondo la nomenclatura stabilita dal CNR. In particolare, mentre il campo **CodAss** è costituito da un codice numerico di identificazione di ogni *record*, assegnato progressivamente dal *driver* del *database* (di nessun interesse per l'utente), il campo **AssCode**, che può anche essere non numerico, rappresenta, in generale, il codice dell'assenza assegnato ufficialmente dal CNR (per esempio, l'assenza per "missione" ha codice **92**). Inoltre, in ogni *record*, il campo **Descr** contiene la frase di descrizione del tipo di assenza, mentre il campo **DescrHy** contiene la stessa frase con le parole spezzate in sillabe, tramite il carattere "p" (chiariremo a cosa serve questo campo quando descriveremo, nel programma **Assenze**, le schermate dei prospetti giornalieri di presenza e assenza dei dipendenti). Infine, il campo **Preced** è costituito da un codice numerico che stabilisce la precedenza nell'ordine di visualizzazione delle colonne (da sinistra a destra) nei prospetti giornalieri delle presenze e assenze; questo codice, la cui utilità sarà meglio chiarita più avanti, può assumere valori compresi fra **99** (massima precedenza) e **0** (*default*).

Tornando al campo **AssCode**, abbiamo detto che questo rappresenta il codice di assenza CNR "in generale" (cioè non sempre): infatti le procedure prevedono l'utilizzo di particolari *records* della tabella, con codici "inventati" da noi, che si aggiungono a quelli ufficiali. Questi *records*, la cui esistenza è indispensabile sin dall'inizio per il corretto funzionamento dei programmi, sono tutti già inseriti nella tabella **CodiciAssenze** del *database* **Proto.mdb**, che si presenta nel modo seguente:

CodAss	Descr	DescrHy	AssCode	Preced
1	PRESENZA	PRESENZA	00	99
2	FERIE	FERIE	FE	90
3	FERIE ANNO PRECEDENTE FRUITE NEL SECONDO SEMESTRE DELL'ANNO SUCCESSIVO	FE/RIE ANNO PRE/CE/DEN/TE FRU/I/TE NE	37	0
4	RIPOSO COMPENSATIVO SPECIALE	RI/PO/SO COMPEN/SATI/VO SPE/CIA/LE	RC	0
5	SANTO PATRONO	SANTO PATRONO	SP	0

Come si può notare, la tabella contiene 5 *records* precostituiti: di questi, solo il numero 3, con codice **37** e precedenza **0**, ha un codice ufficiale del CNR; la sua descrizione completa è: "FERIE ANNO PRECEDENTE FRUITE NEL SECONDO SEMESTRE DELL'ANNO SUCCESSIVO" (con permesso speciale). Gli altri 4 codici, creati ad uso dei programmi, hanno il seguente significato:

- **Codice 00** : indica la presenza del dipendente nella giornata in esame (precedenza **99**);
- **Codice FE** : indica che il dipendente è genericamente "in ferie" (precedenza **90**); i programmi si occupano di controllare la situazione e di determinare ogni volta il codice di assenza "ufficiale", sulla base dell'assunto che un dipendente tende prima ad esaurire le ferie dell'anno precedente entro il primo semestre (codice **31**), poi le

ferie sostitutive delle festività sopresse, fruibili entro l'anno (codice **94**), e infine quelle dell'anno corrente, che può riportare al primo semestre dell'anno successivo (codice **32**); inoltre i programmi controllano che un dipendente, dichiarato in ferie, ne abbia effettivamente diritto, cioè non abbia già esaurito i giorni di ferie che gli spettano; la totale automatizzazione della gestione delle ferie é uno dei punti qualificanti di queste procedure, in quanto libera l'operatore, addetto all'inserimento dei dati giornalieri nel computer, dall'onere di controllare ogni volta la situazione delle ferie di ogni dipendente;

- **Codice RC** : indica che il dipendente è assente per compensare turni di lavoro notturno o altri impegni eccezionali (precedenza **0**); questo codice é stato introdotto per memorizzare, in caso di accordi particolari con la direzione e solo ad uso interno, i casi di assenza per "riposo compensativo speciale" (quello "ordinario" é invece regolarmente previsto e ha codice **91**); nei documenti ufficiali questo tipo di assenza non viene registrata (é considerata "presenza");
- **Codice SP** : indica l'assenza per la festività del "Santo Patrono" (precedenza **0**); l'introduzione di questo codice si é resa necessaria per il fatto che molti Istituti sono frazionati in sedi diverse, con festività locali differenti: un'assenza per **SP** non é conteggiata fra le assenze nei prospetti riassuntivi, ma serve per decrementare di un'unità il numero totale di giorni lavorativi nel periodo di tempo in esame; viceversa, i giorni che (a parte i sabati e le domeniche) sono festivi per tutti, non vanno registrati, neppure individualmente, come giorni di assenza, ma devono essere inseriti, come vedremo, nella tabella **Festivi**;

Per concludere, raccomandiamo vivamente di non accedere mai alla tabella **CodiciAssenze** tramite **Access**, ma sempre tramite il programma che la gestisce, cioè **Adsenze**, che é dotato di apposite procedure di protezione dei dati, impedendo l'esecuzione di operazioni che potrebbero rovinare la tabella.

2 - Tabella "Dipendenti"

Dipendenti : Tabella			
	Nome campo	Tipo dati	Descrizione
PK	CodDip	Contatore	Codice Dipendente
	Name	Testo	Cognome e Nome dipendente
	DataAss	Numerico	Data Assunzione
	DataCess	Numerico	Data Cessazione
	Laureato	Si/No	Serve per il nuovo contratto
	Nord	Numerico	Numero d'ordine dei records

La tabella contiene un *record* per ogni dipendente in servizio presente o passato (ovviamente non é necessario che comprenda i dipendenti cessati dal servizio prima della data iniziale di utilizzo delle procedure). I suoi *campi* hanno significato abbastanza chiaro:

- **CodDip** é il solito indispensabile *campo-chiave*, costituito da un numero progressivo identificatore del record, assegnato dal sistema;
- **Name** contiene cognome e nome del dipendente (o, se si vuole, una qualunque altra stringa di identificazione);
- **DataAss** rappresenta la data di assunzione del dipendente; non é necessario precisarla se il dipendente ha preso servizio prima dell'installazione delle procedure: in

questo caso il programma di gestione della tabella (**Anaper**) assume che il dipendente sia stato assunto il giorno 1/1/1996 (assunzione di *default*);

- **DataCess** rappresenta la data di cessazione dal servizio; non é necessario precisarla se non si prevede che l'evento accada entro l'anno in corso: in questo caso **Anaper** assume che il dipendente cesserà dal servizio il giorno 31/12/2020 (cessazione di *default*);
- **Laureato** é un *campo booleano* (SI o NO): distingue i dipendenti fino al III livello (SI) dagli altri (NO); é stato introdotto in previsione del nuovo contratto dei ricercatori, che sembra ammettere una diversa data di scadenza per la fruizione delle ferie dell'anno precedente;
- **Nord** é un *campo numerico* che rappresenta il numero d'ordine del dipendente nella visualizzazione delle righe (dall'alto al basso) nei prospetti giornalieri delle presenze e assenze; in tutti gli altri casi i dipendenti sono elencati in ordine alfabetico, ma nei prospetti giornalieri potrebbe essere più comodo per l'operatore accedere a dati ordinati in modo diverso (per esempio, con i dipendenti raggruppati per sede, in caso di sedi diverse).

Nel *database Proto.mdb* la tabella contiene un unico *record*, con **Name** = “NON ESISTE”; questo strano record (da non modificare né cancellare) non viene mai visualizzato, ma serve al programma **Anaper** esclusivamente per ragioni tecniche di programmazione. Dimenticarsi della sua “esistenza” !

La tabella **Dipendenti** dovrebbe essere usata, come la precedente, soltanto dai programmi, nello specifico da **Anaper**; tuttavia, nel caso improbabilissimo che sia necessario gestirla direttamente con **Access**, e in particolare debbano essere inseriti o modificati i valori della data nei *campi DataAss* e **DataCess**, é bene che l'utente conosca il formalismo generale **Microsoft** di memorizzazione delle date. Come si può desumere osservando la figura che descrive i *campi* della tabella, il “Tipo dati” dei *campi DataAss* e **DataCess** é definito “Numerico”: infatti i “tempi” (comprensivi della data, delle ore, dei minuti e dei secondi) sono sempre rappresentati in memoria da numeri interi, che indicano il numero assoluto di secondi trascorsi dalle ore 00:00:00 GMT del giorno 1/1/1970 (nascita della **Microsoft** ???). Poiché le procedure trattano soltanto date, cioè giornate intere e non frazioni di giorni, si é stabilito convenzionalmente che ogni “tempo” che rappresenta una data debba essere indicato dal valore numerico corrispondente alle ore 12:00:00 di quella data. In ogni caso, se ci fosse la necessità di convertire manualmente un “tempo” numerico in giorno-mese-anno o viceversa, l'impresa sarebbe molto ardua, per non dire impossibile; per ovviare a questo inconveniente (che potrebbe verificarsi, ripetiamo, solo nell'eventualità molto improbabile di dovere accedere alla tabella con **Access**), é stato fatto un ulteriore programmino di servizio, che esegue per noi la conversione in entrambi i sensi. Il programma, che si chiama **ConvDate.exe**, é disponibile nella stessa “cartella” **EXPORT** in cui si trovano gli altri programmi e il *database Proto.mdb*.

Per illustrare qual è il modo ottimale di utilizzare il programma **ConvDate**, prendiamo l'esempio di un caso realmente accaduto: il pensionamento di un dipendente dell'IRA, di cui vogliamo conoscere la data corrispondente al valore numerico memorizzato nella tabella:

- mandiamo in esecuzione **ConvDate**: compare una *dialog-box* che per il momento riduciamo ad *icona*;

- attiviamo Access, apriamo il *database* dell'IRA e poi la tabella **Dipendenti**: compare una lista di cui la seguente figura rappresenta una parte:

CodDip	Name	DataAss	DataCess	Laureato	Nord
1	ALBERTAZZI VITTORIO	820494000	1609412400	<input type="checkbox"/>	412
2	AMBROSINI ROBERTO	820494000	1609412400	<input checked="" type="checkbox"/>	20
3	ARBIZZANI BRUNELLA	820494000	1609412400	<input type="checkbox"/>	30
4	BALDESCHI LUCIANO	820494000	870343200	<input type="checkbox"/>	40
5	BERGAMINI ROBERTO	820494000	1609412400	<input checked="" type="checkbox"/>	50

- nel record n.4 (BALDESCHI) selezioniamo, con *doppio-click*, il numero inserito nel campo **DataCess** e lo copiamo nella *Clipboard* con **CTRL-C** :

CodDip	Name	DataAss	DataCess	Laureato	Nord
1	ALBERTAZZI VITTORIO	820494000	1609412400	<input type="checkbox"/>	412
2	AMBROSINI ROBERTO	820494000	1609412400	<input checked="" type="checkbox"/>	20
3	ARBIZZANI BRUNELLA	820494000	1609412400	<input type="checkbox"/>	30
4	BALDESCHI LUCIANO	820494000	870343200	<input type="checkbox"/>	40
5	BERGAMINI ROBERTO	820494000	1609412400	<input checked="" type="checkbox"/>	50

- adesso ripristiniamo **ConvDate** (all'inizio il programma mostra la data corrente e il numero intero di secondi che le corrisponde):

CodDip	Name	DataAss	DataCess	Laureato	Nord
1	ALBERTAZZI VITTORIO	820494000	1609412400	<input type="checkbox"/>	412
2	AMBROSINI ROBERTO	820494000	1609412400	<input checked="" type="checkbox"/>	20
3	ARBIZZANI BRUNELLA	820494000	1609412400	<input type="checkbox"/>	30
4	BALDESCHI LUCIANO	820494000	870343200	<input type="checkbox"/>	40
5	BERGAMINI ROBERTO	820494000	1609412400	<input checked="" type="checkbox"/>	50
6	BONDI				60
7	BORTO				414
8	BRAND				80
9	CATTA				416
10	CESAR				100
11	DALLA				110
12	FERET				120
13	FICARI				130
14	GALLE				140
15	GIOIA				150
16	LARI				160
17	MACCAFERRI ANDREA	820494000	1609412400	<input type="checkbox"/>	418

CONVERTE LE DATE IN NUMERI E VICEVERSA

Data: 31 / 12 / 1997 MERCOLEDI'

Numero di secondi 883566000

trascorsi dalle 00:00:00 GMT dell'1/1/1970

- selezioniamo, con *doppio-click*, il numero inserito nella casella in basso della *dialog-box* e lo sostituiamo, premendo **CTRL-V**, con il valore trasferito dalla tabella nella *Clipboard*; notiamo che anche il contenuto della casella superiore cambia e mostra la data di cessazione dal servizio del collega Luciano Baldeschi:

Dipendenti : Tabella						
CodDip	Name	DataAss	DataCess	Laureato	Nord	
1	ALBERTAZZI VITTORIO	820494000	1609412400	<input type="checkbox"/>	412	
2	AMBROSINI ROBERTO	820494000	1609412400	<input checked="" type="checkbox"/>	20	
3	ARBIZZANI BRUNELLA	820494000	1609412400	<input type="checkbox"/>	30	
4	BALDESCHI LUCIANO	820494000	870343200	<input type="checkbox"/>	40	
5	BERGAMINI ROBERTO	820494000	1609412400	<input checked="" type="checkbox"/>	50	
6	BONDI				60	
7	BORTO				414	
8	BRAND				80	
9	CATTA				416	
10	CESA				100	
11	DALLA				110	
12	FERET				120	
13	FICAR				130	
14	GALLE				140	
15	GIOIA				150	
16	LARI				160	
17	MACCAFERRI ANDREA	820494000	1609412400	<input type="checkbox"/>	418	

CONVERTE LE DATE IN NUMERI E VICEVERSA

Data: GIOVEDI'

Numero di secondi

trascorsi dalle 00:00:00 GMT dell'1/1/1970

In generale il programma **ConvDate** (che può essere usato anche per conoscere il giorno della settimana corrispondente a una certa data) trasforma automaticamente il contenuto di una delle due caselle quando l'operatore esegue una qualunque modifica nell'altra. L'operazione inversa a quella illustrata nel nostro esempio produce risultati corretti solo se la data inserita rispetta alcune regole, che abbiamo fissato convenzionalmente e che valgono non solo per **ConvDate**, ma anche per tutti gli altri programmi del pacchetto. E' opportuno a questo punto aprire una parentesi per elencare tali regole, a cui l'operatore deve attenersi ogni volta che il programma richiede l'inserimento di una data:

- la data deve essere costituita da tre numeri, che rappresentano, nell'ordine, il giorno, il mese e l'anno;
- il giorno e il mese devono essere di 1 o 2 cifre (e compresi nei rispettivi intervalli di validità), l'anno deve essere di 4 cifre (siamo vicini al 2000, non possiamo permetterci di indicare l'anno con solo due cifre!);
- i tre numeri possono essere separati da *blank*, *trattino* (-) o *slash* (/);
- la data deve essere compresa fra i giorni 1/1/1970 e 31/12/2037 (o entro intervalli più brevi, secondo la particolare esigenza del programma).

Per concludere, riportiamo l'esempio di una data inserita correttamente, con il conseguente calcolo corretto del giorno della settimana e del numero di secondi (in caso contrario i due campi vengono "abblencati"):



notare l'ulteriore scritta che compare sulla destra della casella inferiore, per indicare che il numero calcolato corrisponde esattamente alle ore 12:00:00 del giorno 1/4/98.

3 - Tabella "Festivi"

Nome campo	Tipo dati	Descrizione
Anno	Numerico	
Mese	Numerico	
Giorno	Numerico	

La tabella contiene un *record* per ogni giorno non lavorativo (esclusi i sabati e le domeniche che sono riconosciuti automaticamente dal programma). Il significato dei tre *campi* (**Anno**, **Mese** e **Giorno**) é ovvio. Notare che in questo caso la data non é espressa nel formato numerico standard (si tratta dell'unica eccezione in tutte le procedure): questo perché i *records*, a differenza degli altri casi, non devono essere generati dai programmi, ma inseriti direttamente dall'operatore, tramite Access, e sarebbe veramente scomodo utilizzare il formato numerico (anche con l'aiuto di **ConvDate**); spetterà al programma **Adsenze** (che é l'unico a usare questa tabella) copiare i *records* in una propria tabella interna, trasformando le date in numeri assoluti di secondi.

E' necessario che la tabella venga aggiornata all'inizio di ogni anno (o anche prima, se si vuole), con l'elenco di tutti i (pochissimi, sfortunatamente!) giorni festivi dell'anno. Nel *database Proto.mdb* la tabella contiene i giorni festivi degli anni 1997 (secondo semestre) e 1998, come si vede nella prossima figura:

Anno	Mese	Giorno
1997	8	15
1997	12	8
1997	12	25
1997	12	26
1998	1	1
1998	1	6
1998	4	13
1998	5	1
1998	12	8
1998	12	25

La tabella dovrà essere aggiornata a partire dal 1999.

4 - Tabella "Movimenti"



Nome campo	Tipo dati	Descrizione
Day	Numerico	Data del giorno
CodDip	Numerico	Codice Dipendente
CodAss	Numerico	Codice presenza/assenza

La tabella contiene un *record* per ogni giornata di presenza o assenza di un dipendente; il *campo Day* rappresenta la data del giorno in esame, nel formato numerico standard, mentre i *campi CodDip* e *CodAss* rappresentano i codici di identificazione rispettivamente del dipendente, ricavato dalla tabella **Dipendenti**, e del tipo di assenza (o della presenza), ricavato dalla tabella **CodiciAssenze**.

La tabella é completamente gestita dai programmi e non dovrebbe essere mai necessario aprirla con **Access**.

Ovviamente il *database Proto.mdb* non contiene *records* della tabella **Movimenti**, né delle due tabelle seguenti.

5 - Tabella "Provvisori"

La struttura di questa tabella é identica a quella della tabella **Movimenti**. Quanto ai contenuti, la differenza fra le due tabelle verrà spiegata più avanti, nella descrizione del programma **Adsenze**.

6- Tabella "FerieInfo"

Dato che questa tabella é molto "complicata", rinunciamo in questa sede a descriverne il contenuto (chi volesse conoscere il significato dei suoi *campi*, può consultare il già citato **Rapporto IRA 239/97**, pagg.24-25). Basti sapere che la tabella contiene informazioni sulle ferie dei dipendenti e si divide in due parti:

- la prima parte (gestita dal programma **Anaper**) è costituita da un *record* per ogni dipendente e per ogni anno: i dati di ciascun *record* riguardano i giorni di ferie che il dipendente è autorizzato a fare nell'anno, con codici **94** ("festività soppresse", normalmente 4 giorni) e **32** ("ferie anno corrente", normalmente 28 giorni);
- la seconda parte (gestita dal programma **Adsenze**) è costituita da un *record* per ogni giornata di ferie fruita da ogni dipendente: contiene informazioni sulla data della giornata di ferie e sul codice CNR del tipo di ferie, calcolato dal programma sulla base del totale delle ferie fruito fino a quel giorno; in pratica, le assenze per ferie, memorizzate nelle tabelle **Movimenti** e **Provvisori** con codice "generico" **FE**, vengono riportate in questa tabella, dopo che il programma le ha automaticamente suddivise secondo i codici "burocratici" del CNR.

Nel caso di eventi imprevisti che richiedano la necessità di correzioni, non bisogna usare **Access**, ma ricorrere al programma **Transfer**, che, come vedremo, riallinea il contenuto della tabella **FerieInfo** ai dati delle tabelle **Movimenti** e **Provvisori**.

Caratteristiche generali dei programmi

In questo capitolo anticipiamo alcune regole, caratteristiche e convenzioni, che sono comuni a più programmi. Nella descrizione di ogni singolo programma non avremo la necessità di ripeterle.

- Interfaccia-utente. La forma grafica e il *design* dell'*interfaccia-utente* sono tali da rendere il lavoro dell'operatore il più semplice e rapido possibile: in particolare l'inserimento dei dati viene sempre esplicitamente richiesto dal programma (quando il programma sta lavorando per conto suo appare una "clessidra" o la scritta: "ATTENDERE, PREGO") e viene sempre mostrato un valore di *default*, coincidente con il dato che si presenta con maggiore frequenza e che quindi necessita il più delle volte soltanto di essere confermato; si permette molto spesso il ricorso alla tastiera come alternativa al *mouse*, quando l'operazione può essere in questo modo più rapida (tasti *acceleratori*); nel caso di visualizzazione di tabelle, con righe e colonne, che occupano un'area maggiore di quella dello schermo (come in **Anaper** e in alcune operazioni di **Adsenze**), lo stesso schermo è spesso suddiviso in più parti, in modo che le intestazioni delle righe e delle colonne rimangano sempre visibili e solidali con le rispettive caselle, mentre il corpo della tabella viene fatto scorrere orizzontalmente o verticalmente (*scrolling*); le "finestre" sono sempre a schermo intero (salvo quella di **Transfer**): è consentita la riduzione a *icona* (come sospensione temporanea del programma per fare altre cose), ma non la riduzione a dimensioni intermedie, per non perdere importanti dettagli dell'informazione; infine, allo scopo di minimizzare la probabilità di errore, abbiamo stabilito che si può terminare il programma sempre in un unico modo: *clickando* sulla casella X in alto a destra, casella che viene opportunamente "oscurata" quando non è consentito interrompere un'operazione in corso. Riassumendo, il principale scopo è stato quello di costruire un'*interfaccia-utente* "amichevole" e di proteggere il più possibile i dati da errori operativi.

- Inserimento delle date. Quando il programma richiede di inserire una data, bisogna attenersi alle regole elencate a pag. 12.

- Prospetti temporanei e definitivi. La parola "Prospetto" viene usata in questo contesto per indicare un'insieme di dati (*records*) che si riferiscono a una giornata lavorativa; ogni *record* di un Prospetto contiene il codice di un dipendente, il codice di una assenza (o della "presenza") e la data della giornata in esame (che è sempre la stessa nello stesso Prospetto e viene detta "data del Prospetto"); quindi un Prospetto descrive, nella giornata a cui si riferisce, la condizione di presenza o il tipo di assenza di ogni dipendente.

Un Prospetto può essere *definitivo* o *temporaneo*: è *definitivo* quando sono verificate entrambe le seguenti condizioni:

- 1) il Prospetto è completo, cioè contiene i *records* relativi a tutti i dipendenti in servizio (sono considerati "in servizio" quei dipendenti che hanno data di assunzione anteriore alla data del Prospetto e data di cessazione posteriore);

2) il Prospetto contiene dati tutti validi, in particolare per quello che riguarda le ferie: ogni dipendente dichiarato in ferie non deve avere esaurito i giorni di ferie che gli spettano.

Un Prospetto nasce sempre come *temporaneo* e viene trasformato automaticamente dal programma in *definitivo* solo quando le due condizioni suddette sono verificate. Tuttavia anche un Prospetto *definitivo* può essere sempre riaperto e modificato, per dare modo all'operatore di correggere gli eventuali errori (anzi, è anche prevista la possibilità che un Prospetto venga ritrasformato da *definitivo* a *temporaneo* se le correzioni introdotte fanno mancare, anche per un solo dipendente, le due condizioni fondamentali). Tutte le trasformazioni dei Prospetti da *temporanei* a *definitivi* (o viceversa) vengono sempre notificate dal programma tramite opportuni messaggi.

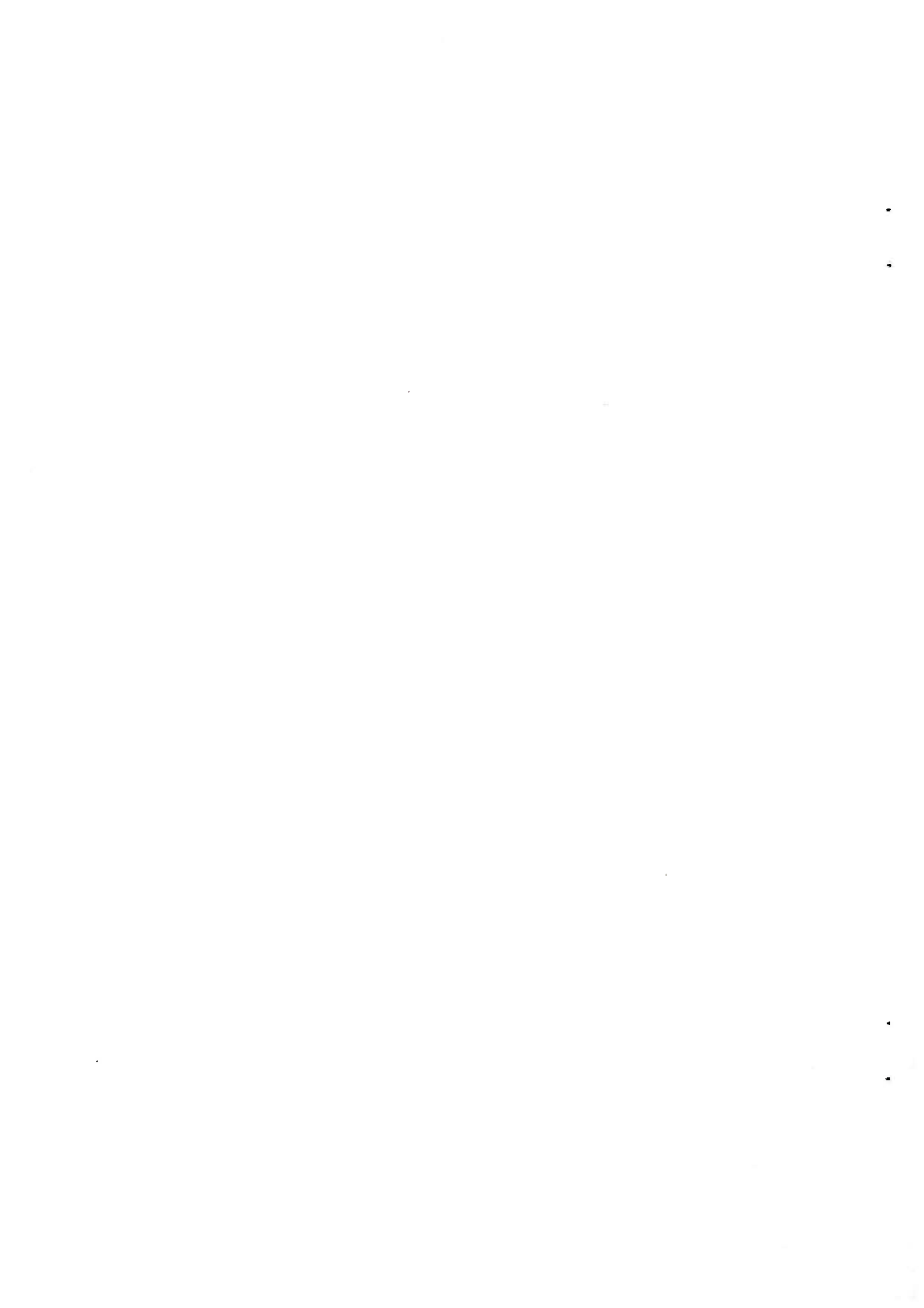
La distinzione fra Prospetti *definitivi* e *temporanei* è importante come misura di salvaguardia per la correttezza dei dati, soprattutto per quello che riguarda la preparazione dei riepiloghi ufficiali: per esempio il programma accetta di preparare il rapporto mensile da inviare al CNR solo se trova i Prospetti, tutti definitivi, di tutte le giornate lavorative del mese. Inoltre la possibilità di preparare provvisoriamente Prospetti incompleti (e perciò *temporanei*) consente di inserire i dati relativi alle cosiddette "previsioni" di assenza: se si prevede che un dipendente resterà assente per un lungo periodo di tempo (per esempio, per maternità, per missione o per ferie), si può, con una sola operazione, creare un grande numero di Prospetti *temporanei* (ognuno con un solo *record*); ogni volta che l'operatore aprirà un Prospetto giornaliero per inserire i dati relativi agli altri dipendenti, troverà già pronta l'informazione introdotta come "previsione", con notevole risparmio di fatica e minore possibilità di errori.

I *records* dei Prospetti *definitivi* vengono memorizzati nella tabella **Movimenti** del *database*, mentre quelli dei Prospetti *temporanei* sono scritti nella tabella **Provvisori** (e da questa cancellati e trasferiti nella tabella **Movimenti** ogni volta che un Prospetto viene trasformato in *definitivo*).

- Giorni festivi. I programmi considerano automaticamente "festivi" tutti i sabati, le domeniche e i giorni indicati nella tabella **Festivi**; non vengono creati Prospetti relativi a giornate "festive"; nei riepiloghi e nei conteggi delle giornate lavorative i giorni "festivi" vengono "saltati".
- Ferie. L'argomento è già stato introdotto durante la descrizione della tabella **CodiciAssenze** del *database* (vedere pagg. 8-9). Nella preparazione dei Prospetti giornalieri (con **Adsenze**) appare sempre un unico codice di ferie (**FE**) (con grande sollievo dell'operatore che non deve preoccuparsi ogni volta di quale tipo di ferie si tratti), mentre nelle visualizzazioni e nei riepiloghi (in **Adsenze** e **Dipcheck**) sono mostrati i tre codici di ferie (**31**, **94** e **32**) regolarmente previsti: la ripartizione è eseguita in modo automatico dal programma (correttamente anche per il futuro nel caso di modifiche in date precedenti) e l'utente non deve mai, per nessuna ragione, introdurre questi codici separatamente.
- Stampe. I programmi, che sono in grado di produrre complessivamente 6 diversi tipi di stampe (1 di **Anaper**, 3 di **Adsenze** e 2 di **Dipcheck**), non utilizzano mai direttamente le stampanti, ma creano, ogni volta, un file di puro testo (estensione

.txt). Per evitare ambiguità, il nome del file è creato automaticamente, componendo insieme il giorno, l'ora, i minuti e i secondi ricavati dal *clock* del Personal Computer (per esempio, se l'operazione è eseguita alle ore 10:23:45 del giorno 30, il nome del file sarà: **30102345.txt**). **Anaper** e **Adsenze** creano il file nella stessa *directory* in cui risiedono, mentre **Dipcheck**, che, essendo utilizzabile da un utente qualsiasi, non potrebbe scrivere nella *directory* dei programmi, crea il file in **C:** (la *root-directory* esiste certamente in ogni PC). Una volta creato il file, il programma ne notifica il nome all'utente in una *box* di messaggio; l'operatore ne prenderà nota e, ritornato alla fine del programma in ambiente **Windows**, potrà cambiare nome al file se vuole conservarlo, oppure, per eseguire la stampa, aprirlo con un'*utility* di **Windows** (per esempio **notepad**) e *clicare* sul comando **Print**.

- **NOTA FINALE IMPORTANTE** : si è riscontrato che ogni tanto, dopo che un programma è terminato, **Windows** produce un messaggio di errore di sistema: il fenomeno (comune a tutti i programmi del pacchetto) è assolutamente inspiegabile, in quanto accade in modo randomico (in media una volta su dieci) ed è apparentemente scorrelato con il tipo di operazioni effettuate precedentemente (a volte si manifesta anche se il *database* è stato usato in sola lettura, cioè in uscita da **Dipcheck**); presumiamo che sia dovuto a qualche "baco" del sistema, o del *driver ODBC*. Comunque abbiamo verificato che questo errore non produce danni al *database* e quindi i dati modificati dal programma possono essere conservati senza preoccupazione. Per quello che riguarda **Windows**, si è riscontrato, qualche volta, un temporaneo "oscuramento" dello schermo; in tali circostanze è buona norma prudenziale terminare immediatamente la sessione di **Windows** e riapirla.



Programma “Anaper”

1 - Considerazioni generali

Il programma **Anaper** gestisce la tabella **Dipendenti** e la parte generale della tabella **FerieInfo**. E' il primo programma che deve essere usato dopo l'installazione delle procedure, in quanto le due tabelle devono essere completate con tutte le informazioni necessarie prima che si cominci ad utilizzare **Adsenze** per l'inserimento dei Prospetti giornalieri. Successivamente **Anaper** sarà usato molto di rado e, precisamente, nelle seguenti circostanze:

- 1) in caso di nuove assunzioni;
- 2) in caso di cessazioni dal servizio entro l'anno in corso, dovute a contratti di lavoro a tempo definito (per esempio, trimestrali);
- 3) appena si giunge a conoscenza della data di pensionamento di un dipendente entro l'anno in corso; in questo e nel caso precedente si dovrà correggere non solo la data di cessazione dal servizio, ma anche il numero di giorni di ferie spettanti al dipendente: tale numero va calcolato, secondo una tabella distribuita dal CNR, in funzione della frazione di anno di servizio del dipendente;
- 4) all'inizio di ogni anno, limitatamente ai casi di dipendenti che abbiano diritto a un numero ridotto di giorni di ferie (per esempio, secondo la nuova normativa, i tecnici assunti da meno di tre anni,); infatti il programma assume di *default* che a ogni dipendente spettino ogni anno 4 gg. di ferie con codice **94** (festività soppresse) e 28 gg. con codice **32** (ferie anno corrente) e quindi é necessario intervenire solo nei casi il cui il numero di giorni é inferiore;
- 5) in generale, per correggere precedenti errori di inserimento dei dati.

In pratica **Anaper** é un programma che impegna considerevolmente l'operatore soltanto la prima volta: ma questa prima volta bisognerà fare molta attenzione, soprattutto per quello che riguarda il corretto inserimento del numero di giorni di ferie spettanti a ogni dipendente (come vedremo, bisogna inserire i dati relativi ai tre codici, **94**, **32** e **31**, separatamente). Infatti, ogni volta che, con **Anaper**, si crea un nuovo dipendente, il programma crede veramente che si tratti di una nuova assunzione e determina i valori di *default* dei giorni di ferie spettanti, che sono:

4 (cod. 94), 28 (cod. 32) , 0 (cod. 31) ;

nel caso reale di nuove assunzioni bisognerà correggere i valori relativi ai codici **94** e **32** in base alle tabelle CNR, mentre il valore relativo al codice **31** non va ovviamente modificato; ma, quando si inseriscono i dati per la prima volta, questi riguardano non “nuovi assunti” ma dipendenti “anziani” che, in generale, devono ancora fruire di giorni di ferie dell'anno precedente; quindi é necessario, per ogni dipendente, calcolare e inserire il valore corretto relativo al codice **31**. Inoltre, se la data iniziale di utilizzo delle procedure non coincide con l'inizio dell'anno (e a meno che non si vogliano inserire i dati retroattivamente dal primo gennaio), bisogna tenere conto, per ogni dipendente, dei giorni di ferie fruite nella parte dell'anno che non viene registrata e inserire i valori dei giorni di ferie rimasti; in altre parole simuliamo che l'anno cominci effettivamente quando inizia la registrazione dei dati, ma che i giorni di ferie spettanti

per tutto l'anno non siano quelli reali, bensì quelli calcolati detraendo i giorni di ferie già fruiti (che vanno sottratti, anzitutto, al valore relativo al codice **31**, poi, se questo si azzerà, al valore relativo al codice **94** e infine, se anche questo si azzerà, al valore relativo al codice **32**). Per esempio, supponiamo che a un dipendente spettino per tutto l'anno i seguenti giorni di ferie:

4 (cod. 94), 28 (cod. 32), 5 (cod. 31) ;

ma che lo stesso dipendente abbia già fruito, nella parte dell'anno che precede la data iniziale delle registrazioni, di 7 giorni di ferie; pertanto i valori corretti da inserire con **Anaper** sono:

2 (cod. 94), 28 (cod. 32), 0 (cod. 31) .

2 - Avvio del programma

Quando si lancia il programma **Anaper**, appare, dopo qualche secondo, la seguente *dialog-box*:



L' "anno di riferimento" (*default*: l'anno corrente) é importante soprattutto per quello che riguarda il significato dei tre valori di giorni di ferie spettanti (i concetti di "ferie anno corrente" e di "ferie anno precedente" sono relativi, appunto, all'anno di riferimento), ma anche perché serve per selezionare dal *database* la lista dei dipendenti su cui il programma deve lavorare: da questa lista sono infatti esclusi i dipendenti assunti dopo il 31/12 e quelli cessati dal servizio prima dell'1/1. L'operatore può inserire un valore dell'anno non inferiore al 1997 e non superiore all'anno corrente (in pratica, per ora, solo il 1997 o il 1998). I nuovi utilizzatori delle procedure dovranno lasciare invariato il valore di *default*, *clickando* direttamente sul "bottone" **OK** oppure premendo il tasto **Enter** (di solito nelle *dialog-box* il tasto *acceleratore* **Enter** é associato al "bottone" **OK**, ma talvolta non é così, soprattutto quando si tratta di operazioni "delicate" che richiedono un attimo di ripensamento; segnaleremo, d'ora in poi, solo questi casi).

3 - Descrizione della finestra di schermo

Una volta selezionato l'anno di riferimento, il programma utilizza questa informazione per estrarre dalle tabelle **Dipendenti** e **FerieInfo** i dati che gli servono, e per organizzare la loro presentazione sullo schermo (per queste operazioni impiega alcuni se-

condi). Ovviamente ciò che si vedrà dipende dalla consistenza del *database*: quelli che utilizzano **Anaper** per la prima volta vedranno uno schermo praticamente vuoto, con le sole intestazioni; noi invece presenteremo, nelle prossime illustrazioni di esempio, la tabella del *database* dell'IRA, che è molto consistente e si estende ben oltre le dimensioni dello schermo stesso (può essere vista per intero solo con lo *scrolling*).
 terminate le operazioni preliminari, la finestra di schermo assume il seguente aspetto (nella figura ne riportiamo solo una parte):

ANNO DI RIFERIMENTO: 1998		Num.	Data	Data	Ferie spettanti			Delete	
Codice	COGNOME e NOME	Laur.	Ordine	Assunzione	Cessazione	(gg. / codice)			Record
						94	32	31	
042	ZACCHIROLI GIAMPAOLO		0432	default	default	4	28	17	Canc
043	ZINI IADER		0434	default	default	4	28	9	Canc
044	BUTTACCIO TARDIO SALVATORE		0440	default	default	4	28	15	Canc
045	CONTAVALLE CORRADO		0450	default	default	4	28	12	Canc
046	LETO GIUSEPPE	X	0460	default	default	4	28	2	Canc
047	NICOTRA GAETANO		0470	default	default	4	28	15	Canc
053	NICOTRA LEONARDO		0472	16-06-1997	default	4	26	11	Canc
054	NOGITA CARLO		0475	16-06-1997	default	4	26	11	Canc
048	PAPALEO LUIGI		0480	default	default	4	28	18	Canc
049	STANGHELLINI CARLO	X	0490	default	default	4	28	17	Canc
050	TUCCARI GINO	X	0500	default	default	4	28	10	Canc
051	TRIGILIO CORRADO	X	0510	default	default	4	28	7	Canc
052	UMANA GRAZIA MARIA GLORIA	X	0520	default	default	4	28	25	Canc
NEW									

Notiamo anzitutto che il programma non ha praticamente *menù*, in quanto, come vedremo, tutte le operazioni si eseguono interagendo direttamente con la finestra di schermo. L'unico comando di *menù* presente (**Frame**) è del tutto privo di importanza (lasciamo all'utente il compito di scoprire a che cosa serve!).

Sullo schermo appare una tabella che riproduce, mediante una opportuna interfaccia di visualizzazione, le due tabelle dal *database*, con i *records* lungo le righe, i *campi* lungo le colonne e i *valori* dei dati nelle *caselle* di intersezione fra righe e colonne. Per certe colonne sono necessari alcuni chiarimenti:

- la prima colonna a sinistra contiene i valori del *campo* **CodDip** (codice progressivo) della tabella **Dipendenti**; sebbene tale *campo* non interessi minimamente all'utilizzatore, la presenza di questa colonna è utile perché le sue caselle, non essendo modificabili (*readonly*), non attivano operazioni e quindi possono servire da "parcheggio" negli spostamenti da una casella all'altra;
- la terza colonna contiene i valori del *campo booleano* **Laureato** ed è vuota se il suo valore è FALSO (livelli IV-X), mentre contiene una **X** se il suo valore è VERO (livelli I-III);
- le righe sono ordinate secondo valori crescenti del *campo* **Nord**, specificato nella quarta colonna;
- la quinta e sesta colonna contengono quasi sempre la scritta "**default**" (vedere pagg. 9-10, *campi* **DataAss** e **DataCess**); è realmente necessario specificare la data di assunzione (o di cessazione) quando questa è compresa entro l'anno di riferimento;

- le successive tre colonne contengono i valori dei *campi* della tabella **FerieInfo**; é bene precisare ancora una volta che tali valori non indicano i giorni di ferie che il dipendente deve ancora fare (salvo la prima volta che si inseriscono i dati, come specificato a pagg. 19-20), ma il totale delle ferie spettanti nell'anno di riferimento;
- l'ultima colonna non rappresenta un *campo* delle tabelle del *database*, ma un "comando", sul quale bisogna *clickare* se si vuole la cancellazione del *record* corrispondente sulla riga (come vedremo, ciò é consentito solo a determinate condizioni);
- l'ultima riga, che appare vuota salvo la scritta NEW in prima colonna, è riservata all'inserimento dei nuovi *records* (cioè dei nuovi dipendenti), come vedremo più avanti..

4 - Selezione delle caselle e spostamenti

Abbiamo detto che il contenuto di ogni *casella* rappresenta il *valore* assunto dal corrispondente *campo* (colonna) del corrispondente *record* (riga) del *database* (a parte le caselle dell'ultima colonna, che per il momento non consideriamo). Durante l'esecuzione del programma, esiste sempre una (e una sola) *casella* il cui colore di fondo è grigio o bianco, mentre il colore di tutte le altre caselle è azzurro e coincide con lo sfondo generale della finestra di schermo. La *casella* che si differenzia dalle altre si chiama "*casella* selezionata". All'inizio la *casella* selezionata è quella con la scritta NEW (prima colonna dell'ultima riga) che appare con fondo grigio. Il colore di fondo di una *casella* selezionata è importante perché indica il tipo di azione che si può eseguire sul contenuto della *casella* e precisamente:

- se la *casella* selezionata appartiene alla prima colonna (codice progressivo) il suo contenuto è *readonly*, cioè non può essere modificato; il colore di fondo è grigio;
- se la *casella* selezionata appartiene alla terza colonna (*campo* di tipo *booleano*) il suo contenuto non può essere modificato, ma solo invertito fra gli stati VERO e FALSO; il colore di fondo è ancora grigio;
- in tutti gli altri casi il contenuto della *casella* selezionata può essere modificato; si dice che la *casella* è "attiva" e il suo colore di fondo è bianco.

Il modo più semplice (ma non sempre il più rapido) per selezionare una *casella* è quello di posizionare su di essa il cursore del *mouse* e poi *clickare*. In certi casi tuttavia (per esempio quando si devono inserire nuovi dati sequenzialmente) può essere più comodo utilizzare la tastiera; ciò è possibile in quanto esistono 9 tasti (che chiamiamo "tasti speciali") adibiti a questo scopo (tutti gli altri tasti servono per operare all'interno di una *casella* "attiva"). Questi tasti sono:

- **Tab** : seleziona la *casella* della colonna successiva sulla stessa riga, oppure, se parte dalla penultima colonna (escludiamo sempre l'ultima, che non è selezionabile da tastiera), la *casella* della prima colonna della riga successiva;
- **Enter** : seleziona la *casella* della prima colonna della riga successiva (può essere usato per "saltare" le caselle della stessa riga che non interessa modificare);
- **↑** : sposta la selezione di una riga in su, stessa colonna;
- **↓** : sposta la selezione di una riga in giù, stessa colonna;
- **PageUp** : sposta la selezione di 10 righe in su, stessa colonna;
- **PageDown** : sposta la selezione di 10 righe in giù, stessa colonna;

- **Home** : sposta la selezione sulla prima riga, stessa colonna;
- **End** : sposta la selezione sull'ultima riga, stessa colonna.
- **Esc** : ha una duplice funzione:
 - 1) se, dopo un'operazione di *scrolling* effettuata dall'operatore, la *casella* selezionata non è più visibile, esegue uno *scrolling* automatico che riporta la *casella* selezionata sullo schermo;
 - 2) se la *casella* selezionata è visibile, ripristina il suo contenuto originario e poi abbandona la *casella* e seleziona quella della prima colonna, stessa riga;
 il tasto **Esc** è molto utile: nel primo caso serve quando l'operatore ha bisogno di visionare un dato che si trova fuori dalla finestra di schermo prima di completare le modifiche da introdurre nella *casella* attiva (e il tasto **Esc** gli permette di ritornare immediatamente dentro la *casella*); nel secondo caso serve quando l'operatore "si pente" di avere modificato il contenuto di una *casella* e vuole uscirne senza che le modifiche vengano trasferite al *database* (caso "classico" di uso del tasto **Esc**).

Quando è selezionata una *casella* "attiva" il programma permette di operare all'interno della *casella* utilizzando tutte le funzioni del *mouse* (selezione del testo mediante "trascinamento" ecc...) e della tastiera (salvo i "tasti speciali"); queste funzioni sono ben note agli utilizzatori abituali dei "word processors" e quindi non ne parleremo; aggiungiamo solo che, *clickando* con il tasto destro del *mouse* dentro una *casella* attiva, si ottiene l'apertura di un *menù* di *pop-up* che permette l'esecuzione di alcune operazioni classiche di *editing*, quali *cut*, *copy*, *paste* ecc...(invece *ctrl-c* e *ctrl-v* stranamente non funzionano), utili, per esempio, se si vuole copiare il contenuto di una *casella* in un'altra.

Un caso particolare è costituito dalle caselle della terza colonna (*campo* di tipo *booleano*) che presentano le seguenti caratteristiche:

- la selezione di una casella con il *mouse* produce anche l'immediata inversione (da VERO a FALSO o viceversa) del suo contenuto;
- lo stesso risultato si ottiene, quando la casella è già selezionata, premendo un qualunque tasto non "speciale" (per esempio, la barra spaziatrice).

Poiché esiste sempre una *casella* selezionata, la *selezione* di una nuova *casella* comporta sempre l'*abbandono* di un'altra *casella*: questa azione attiva una serie di controlli e di operazioni che il programma effettua sul contenuto della *casella abbandonata*, e precisamente:

1. se il contenuto della *casella* non è stato modificato, non fa nulla (cioè si limita a spostare la selezione sulla nuova *casella*);
2. se le modifiche apportate non sono ammissibili (il tipo di controllo dipende dal *campo*, cioè dalla colonna a cui appartiene la *casella*, come vedremo più avanti), notifica l'errore con un messaggio e poi mantiene la selezione all'interno della *casella* (in altre parole il programma si rifiuta di selezionare una nuova *casella* se le modifiche eseguite sulla precedente non sono valide); a questo punto l'operatore può apportare ulteriori modifiche per correggere l'errore, oppure rinunciare, premendo il tasto **Esc** e così ripristinando il contenuto iniziale della *casella*;
3. se i controlli danno esito positivo, trasferisce il contenuto della *casella* nel *database*, e poi sposta la selezione sulla nuova *casella*.

Da quanto detto si deduce che l'interfaccia-utente è on-line con il database, che viene aggiornato ogni volta, nel momento esatto in cui una *casella* è *abbandonata* (e se è stata modificata); si è preferita questa soluzione a quella di un unico aggiornamento finale in quanto spesso l'ammissibilità del contenuto di un *campo* dipende dal contenuto di un altro *campo* e quindi è opportuno che i controlli di validità siano effettuati nel quadro di una situazione sempre aggiornata.

Nel caso particolare che la *casella abbandonata* appartenga alla prima colonna, che è per definizione non modificabile, il programma si ferma al punto 1. (cioè non fa nulla); per questo motivo le *caselle* della prima colonna servono come "parcheggio": vanno selezionate quando si vuole *abbandonare* una *casella* attiva per confermare le modifiche (e quindi trasferirle al *database*) oppure per cancellarle (usando il tasto **Esc**), ma non si vuole attivare una nuova *casella* (per esempio quando si vuole uscire dal programma).

Il programma lavora in modo da garantire che la *casella* selezionata sia sempre visibile sullo schermo, effettuando uno *scrolling* automatico quando è necessario. Se l'operatore esegue uno *scrolling* che porta una *casella* attiva fuori dallo schermo, (allo scopo, per esempio, di selezionare una nuova *casella* che si trova lontano da quella che vuole *abbandonare*), non appena tenta di selezionare la nuova *casella* riceve un messaggio di errore, a cui segue uno *scrolling* automatico che riporta la precedente *casella* dentro lo schermo; per ovviare a questo "inconveniente", deve prima selezionare una *casella* di "parcheggio" (cosa che può fare rapidamente premendo **Enter** per confermare le precedenti modifiche oppure **Esc** per cancellarle) e poi è "libero di andare dove vuole". Questa apparente "complicazione" è stata introdotta come misura di protezione da possibili errori: poiché l'*abbandono* di una *casella* comporta l'aggiornamento immediato del *database*, è opportuno che questa operazione sia fatta in modo "consapevole" e se inavvertitamente è stato premuto un tasto mentre la *casella* attiva da *abbandonare* non è visibile, sarà poi difficile accorgersi dell'errore.

5 - Aggiornamenti e controlli di validità

Abbiamo visto che i controlli sulla validità delle modifiche apportate a una *casella* attiva, che precedono il trasferimento al *database*, avvengono non quando la *casella* è *selezionata*, ma quando la stessa è *abbandonata*, per la selezione di una nuova *casella*, e che il programma si rifiuta di spostare la selezione se i controlli danno esito negativo. Vediamo ora in cosa consistono tali controlli, che elenchiamo percorrendo le colonne, cioè i *campi*..

Cognome e Nome (*campo Name*)

- 1) il programma elimina gli spazi bianchi in testa e in coda, lasciando invariati gli eventuali spazi fra una parola e l'altra; ne deve risultare una stringa con almeno due caratteri;
- 2) non devono esistere duplicati, cioè non deve esistere, nella tabella **Dipendenti**, un altro *record* con la stessa stringa nel *campo Name*; il controllo viene fatto comprimendo temporaneamente gli spazi bianchi fra le parole

Liv. I - III (*campo Laureato*) nessun controllo

Num. Ordine (*campo Nord*)

- 1) il numero deve essere compreso fra **1** e un certo valore massimo, fissato in modo da garantire che gli eventuali successivi nuovi *records* abbiano di *default* valori superiori (cioè, appena creati, restino in fondo alla lista);
se il numero introdotto è valido, la posizione della riga nella lista viene aggiornata immediatamente.

Data Assunzione (*campo DataAss*)

- 1) la data deve essere sintatticamente corretta (vedere regole a pag. 12);
- 2) il dipendente non deve essere stato incluso in Prospetti *definitivi* o *temporanei* con data anteriore (non si può sostenere che un dipendente è stato assunto dopo che è già stato registrato presente o assente in qualche Prospetto!);
- 3) la data non deve essere anteriore a quella di *default* (1/1/1996) ; al suo posto il programma suggerisce di scrivere la parola “**default**” oppure di lasciare una riga vuota (che il programma stesso trasformerà in “**default**”);
- 4) la data non deve essere posteriore al 31/12 dell’anno di riferimento, in quanto il dipendente non si troverebbe correttamente collocato in questa lista (in tal caso conviene rinunciare premendo il tasto **Esc**, uscire dal programma e rientrare con il corretto “anno di riferimento”);
se la data introdotta è valida, questa viene riscritta nel formato standard **xx-xx-xxxx**, comunque sia stata scritta dall’operatore.

Data Cessazione (*campo DataCess*)

- 1) la data deve essere sintatticamente corretta (vedere regole a pag. 12);
- 2) il dipendente non deve essere stato incluso in Prospetti *definitivi* o *temporanei* con data posteriore;
- 3) la data non deve essere posteriore a quella di *default* (31/12/2020) ; al suo posto il programma suggerisce di scrivere la parola “**default**” oppure di lasciare una riga vuota (che il programma stesso trasformerà in “**default**”);
- 4) la data non deve essere anteriore all’ 1/1 dell’anno di riferimento, in quanto il dipendente non si troverebbe correttamente collocato in questa lista (in tal caso conviene rinunciare premendo il tasto **Esc**, uscire dal programma e rientrare con il corretto “anno di riferimento”);
se la data introdotta è valida, questa viene riscritta nel formato standard **xx-xx-xxxx**, comunque sia stata scritta dall’operatore.

Ferie spettanti gg./codice 94 (*campo della tabella FerieInfo* - festività soppresse)

- 1) il numero deve essere compreso fra **0** e **4**

Ferie spettanti gg./codice 32 (*campo della tabella FerieInfo* - ferie anno corrente)

- 1) il numero deve essere compreso fra **0** e **28**

Ferie spettanti gg./codice 31 (*campo della tabella FerieInfo* - ferie anno precedente)

- 1) il numero deve essere compreso fra **0** e **28** ; in realtà questo intervallo viene spostato all’indietro di una quantità pari al numero di gg. di ferie già fruito nell’anno precedente: ne può anche risultare, per assurdo, un numero negativo (che si verifica quando i giorni di ferie dell’anno precedente, registrati in Prospetti ancora *tempo-*

ranei, hanno superato quelli spettanti e quindi nell'anno in corso i gg. di ferie spettanti con codice **31** sono diventati sotto-zero !!!)

6 - Inserimento di un nuovo record

Se l'operatore seleziona una qualunque *casella* dell'ultima riga (salvo quella della prima colonna), segnala al programma che intende inserire nella tabella un nuovo dipendente; il programma chiede conferma: se la risposta è NO, seleziona la casella di "parcheggio" con la scritta NEW; se invece la risposta è SI, crea immediatamente un nuovo *record* nelle tabelle **Dipendenti** e **FerieInfo** del *database*, sposta di una posizione verso il basso la riga vuota e al suo posto inserisce una nuova riga, che contiene i valori di *default* attribuiti ai *campi* del nuovo *record*, come appare nella seguente figura di esempio:

ANNO DI RIFERIMENTO: 1998		Liv.	Num.	Data	Data	Ferie spettanti			Delete
Codice	COGNOME e NOME	I-III	Ordine	Assunzione	Cessazione	(gg. / codice)			Record
						94	32	31	
046	LETO GIUSEPPE	X	0460	default	default	4	28	2	Canc
047	NICOTRA GAETANO		0470	default	default	4	28	15	Canc
053	NICOTRA LEONARDO		0472	16-06-1997	default	4	26	11	Canc
054	NOCITA CARLO		0475	16-06-1997	default	4	26	11	Canc
048	PAPALEO LUIGI		0480	default	default	4	28	18	Canc
049	STANGHELLINI CARLO	X	0490	default	default	4	28	17	Canc
050	TUCCARI GINO	X	0500	default	default	4	28	10	Canc
051	TRIGILIO CORRADO	X	0510	default	default	4	28	7	Canc
052	UMANA GRAZIA MARIA GLORIA	X	0520	default	default	4	28	25	Canc
058	58		0580	default	default	4	28	0	Canc
NEW									

Come si può notare, il valore inserito della seconda colonna coincide con lo stesso codice progressivo (è un "nome" di dipendente che sicuramente non ha duplicati!), mentre il Num. d'ordine è ancora lo stesso codice, moltiplicato per 10 (da cui si arguisce che il valore massimo ammesso prima di creare questo *record* era 579, mentre ora sarà 589 e così via...); sicuramente il contenuto del *campo Name* dovrà essere cambiato, mentre quello del *campo Nord* sarà modificato o meno a secondo della posizione nella lista in cui si intende collocare la riga del nuovo dipendente. Per quello che riguarda gli altri *campi*: **Laureato** è inizialmente FALSO, le date di assunzione e cessazione sono quelle di *default*, mentre i gg. di ferie sono **4-28-0**, come previsto.

7 - Come rendere veloci le operazioni di inserimento sequenziale di nuovi records

Per modificare il contenuto di *caselle* che si trovano in posizioni sparse, il modo più semplice per selezionarle è quello di andarle a cercare *scrollando* la lista (oppure utilizzare i tasti di spostamento "grosso", quali **PageUp**, **PageDown**, **Home** e **End**) e poi *cliccare* con il *mouse*. Ma quando si devono inserire nuovi *records* sequenzialmente (come è necessario la prima volta), oppure si devono modificare *caselle* in posizioni contigue, l'utilizzo della tastiera è molto più pratico e veloce. Vorremmo suggerire

una sequenza di operazioni particolarmente indicate per rendere più rapido e meno stressante il lavoro di inserimento iniziale; queste operazioni escludono completamente l'uso del mouse:

si parte dalla casella con la scritta NEW;

- tasto TAB (appare la richiesta di conferma della creazione di un nuovo *record*)
- tasto Enter (per rispondere SI)
- inserimento Cognome e Nome (non perdere tempo a cancellare il numero di codice: questo sparirà automaticamente appena si comincia a digitare); un altro consiglio: lasciare più di un solo *blank* fra cognome e nome per migliorare l'estetica nelle visualizzazioni sullo schermo (se il cognome termina, come spesso accade, con la lettera "I", il formato di scrittura di tipo proporzionale tende a comprimere lo spazio e quindi a riunire insieme il cognome e il nome);
- tasto TAB (seleziona il *campo Laureato*): se il dipendente è di Livello I-III, premere la barra spaziatrice per invertire da FALSO a VERO, altrimenti passare oltre;
- tasto TAB (seleziona il *campo Nord*): se si inseriscono i *records* già nell'ordine desiderato, non è mai necessario modificare il contenuto di questa *casella*;
- tasto TAB (seleziona il *campo DataAss*): è necessario modificare il contenuto di questa *casella* solo se il dipendente è stato assunto quest'anno;
- tasto TAB (seleziona il *campo DataCess*): è necessario modificare il contenuto di questa *casella* solo se il dipendente cesserà dal servizio entro quest'anno;
- tasto TAB (seleziona ferie codice **94**): è possibile che il contenuto di questa casella debba essere modificato, vedere pagg. 19-20;
- tasto TAB (seleziona ferie codice **32**): come sopra;
- tasto TAB (seleziona ferie codice **31**): come sopra;
- tasto TAB o Enter (seleziona , nella riga successiva, la casella con la scritta NEW): ripeti il ciclo per il prossimo *record*.

8 - Cancellazione di un record

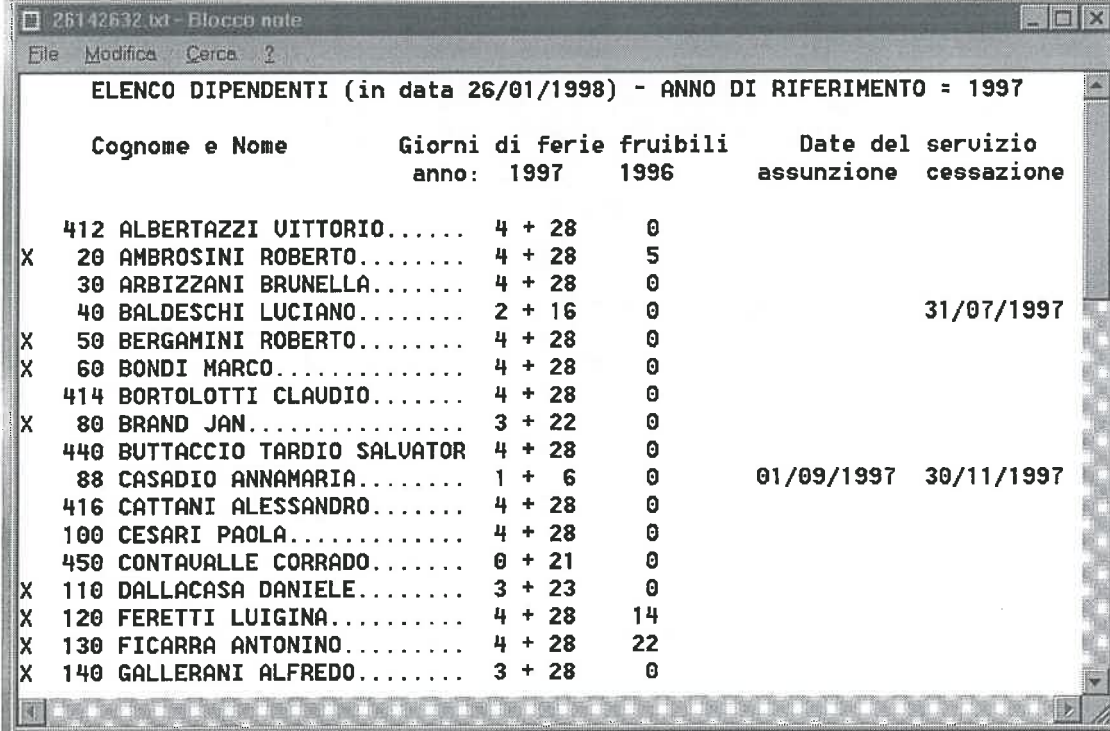
La necessità di cancellare un *record* è un evento estremamente raro (può capitare, per esempio, se un dipendente, nuovo assunto e già inserito in tabella, rinuncia prima di prendere servizio); comunque, per completezza, si è inserito nel programma anche questa operazione. Per tentare di cancellare un *record* bisogna *clikkare* sulla *casella* dell'ultima colonna della riga che corrisponde a quel *record*: in risposta a questo comando il programma cerca il codice del dipendente nelle tabelle **Movimenti** e **Provvisori** e, se lo trova, si rifiuta di eseguire l'operazione, notificando un messaggio di errore, (il dipendente che si vuole cancellare, se è stato inserito per errore in qualche Prospetto, prima deve essere rimosso dal Prospetto (con **Adsenze**), e poi può essere tolto dalla tabella **Dipendenti**); in caso contrario, l'operazione è ammissibile, ma per ulteriore sicurezza, ne viene chiesta conferma: solo se la risposta è SI il programma cancella il *record* nel *database* e corrispondentemente rimuove la riga dalla lista.

La cancellazione di un *record* è un'operazione indipendente dalla selezione di una *casella*, che rimane inalterata; se la *casella* selezionata apparteneva alla riga che è stata rimossa, la selezione passa sulla prima colonna della riga successiva.

9 - Fine del programma e creazione del file di stampa

Se esiste una *casella* attiva nel momento in cui il programma termina, tutte le modifiche apportate al suo contenuto vengono perse, in quanto la *casella* non ha fatto in tempo ad essere *abbandonata* e quindi non ha potuto trasmettere il suo contenuto al *database*. E' quindi conveniente, prima di far terminare il programma, abbandonare una *casella* attiva e selezionare una casella "di parcheggio" (per esempio premendo il tasto **Enter**).

Quando l'operatore *clicca* sulla casella con la **X** del *system menù* (in alto a destra) il programma non termina immediatamente, ma prima chiede all'operatore se vuole creare il file di stampa (in questo caso la "risposta preferenziale", associata al tasto **Enter**, è NO); se l'operatore risponde SI, viene creato un file di testo, che differisce dalla lista sullo schermo per il fatto che i dipendenti sono in ordine alfabetico e le colonne sono arrangiate in modo diverso, come appare dalla seguente figura:



26142632.txt - Blocco note

File Modifica Cerca ?

ELENCO DIPENDENTI (in data 26/01/1998) - ANNO DI RIFERIMENTO = 1997

	Cognome e Nome	Giorni di ferie fruibili		Date del servizio	
		anno: 1997	1996	assunzione	cessazione
	412 ALBERTAZZI VITTORIO.....	4 + 28	0		
X	20 AMBROSINI ROBERTO.....	4 + 28	5		
	30 ARBIZZANI BRUNELLA.....	4 + 28	0		
	40 BALDESCHI LUCIANO.....	2 + 16	0		31/07/1997
X	50 BERGAMINI ROBERTO.....	4 + 28	0		
X	60 BONDI MARCO.....	4 + 28	0		
	414 BORTOLOTTI CLAUDIO.....	4 + 28	0		
X	80 BRAND JAN.....	3 + 22	0		
	440 BUTTACCIO TARDIO SALVATOR	4 + 28	0		
	88 CASADIO ANNAMARIA.....	1 + 6	0	01/09/1997	30/11/1997
	416 CATTANI ALESSANDRO.....	4 + 28	0		
	100 CESARI PAOLA.....	4 + 28	0		
	450 CONTAVALLE CORRADO.....	0 + 21	0		
X	110 DALLACASA DANIELE.....	3 + 23	0		
X	120 FERETTI LUIGINA.....	4 + 28	14		
X	130 FICARRA ANTONINO.....	4 + 28	22		
X	140 GALLERANI ALFREDO.....	3 + 28	0		

notiamo che il valore del *campo Laureato* appare in prima colonna; segue il numero d'ordine, il cognome e nome, i giorni di ferie e, per ultime, le date di assunzione e cessazione, che sono scritte solo quando differiscono dal *default*.

Programma “Adsenze”

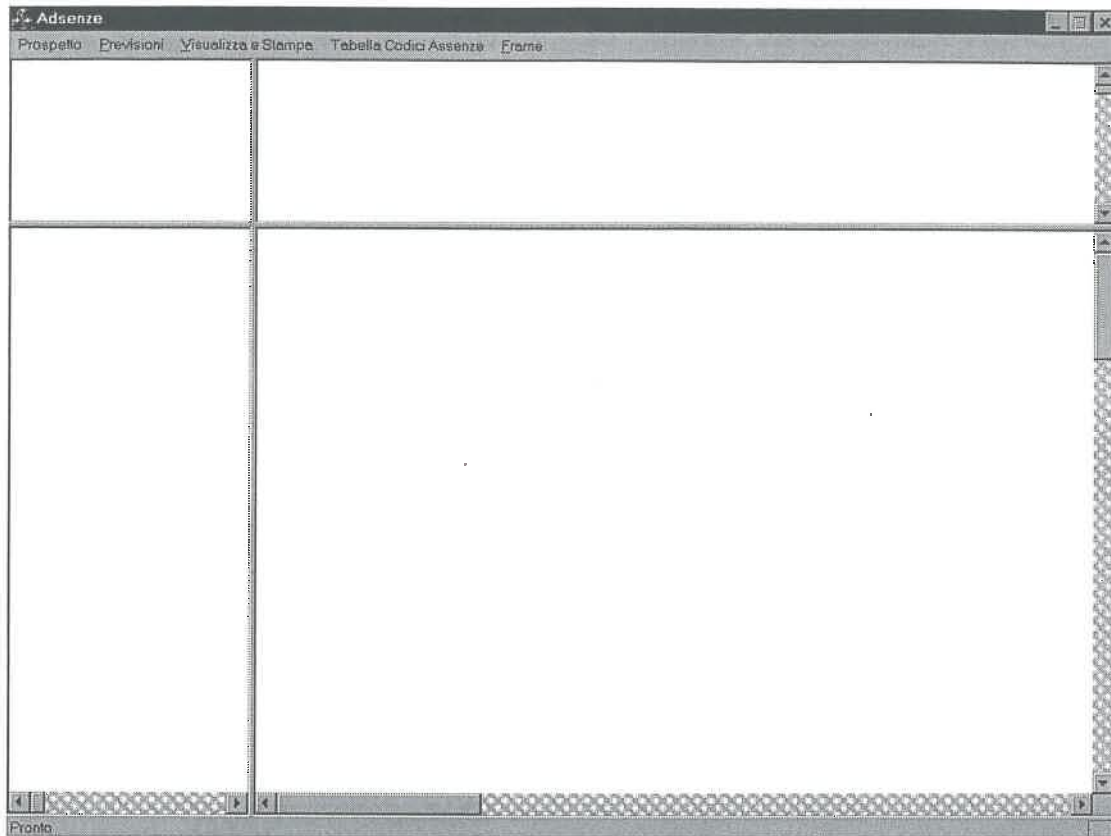
1 - Considerazioni generali

Il programma **Adsenze** è il più importante del pacchetto: va usato quotidianamente per l’inserimento dei Prospetti giornalieri delle presenze/assenze dei dipendenti nelle tabelle **Movimenti** e **Provvisori** del *database*; inoltre consente di inserire, con una sola operazione, le “previsioni” di assenza di un dipendente per un intero periodo di tempo (vedere pag. 16); prepara e mostra i riepiloghi statistici con il numero dei gg. di presenza/assenza, e il rapporto mensile da trascrivere sull’apposito modulo da inviare al CNR; infine gestisce la tabella **CodiciAssenze**. Per queste operazioni il programma usa, in sola lettura, le tabelle **Dipendenti**, **Festivi** e la parte generale della tabella **FerieInfo**, mentre utilizza la stessa tabella **FerieInfo** per memorizzare i giorni di ferie fruiti dai dipendenti, automaticamente suddivise, a partire dal codice “generico” **FE** memorizzato nei Prospetti, nei tre codici “burocratici” **94**, **32** e **31**. La gestione delle ferie si estende anche al caso particolare (e molto raro) del codice **37** (ferie dell’anno precedente fruiti, con permesso speciale, nel secondo semestre dell’anno successivo): anche per questo tipo di ferie i dati sono trascritti nella tabella **FerieInfo** e viene controllato che il numero di gg. non superi quello spettante (mantenendo nello stato di *temporanei* i Prospetti in cui ciò si dovesse verificare).

Per quanto riguarda la tabella **CodiciAssenze**, il programma suddivide la sua gestione in due parti: le operazioni di modifica e cancellazione dei *records*, insieme a quella di stampa del contenuto della tabella, vengono eseguite a parte e sono attivate da un comando di *menù* adibito a questo scopo; invece la creazione di nuovi *records* è contestuale all’operazione di inserimento dei Prospetti: l’operatore è tenuto a creare un nuovo *record* solo quando ne ha bisogno, cioè solo quando si verifica un caso reale di assenza, il cui codice non è ancora stato registrato nella tabella. Infatti esiste una profonda differenza fra le tabelle **Dipendenti** e **CodiciAssenze**: la prima deve essere già pronta e completa prima di cominciare (come abbiamo visto, si deve usare il programma **Anaper**), mentre per la seconda è sufficiente partire con i soli *records* precostituiti, forniti nel *database* **Proto.mdb** (fra cui sono veramente importanti quelli con codici **00** e **FE**). Questa soluzione è stata adottata, come al solito, per rendere più comodo e pratico il lavoro: infatti i codici di assenza previsti dal CNR sono più di un centinaio e sarebbe veramente oneroso inserirli tutti in anticipo (considerato anche che molti di essi non saranno mai utilizzati), e quindi ci è sembrato molto più efficiente e pratico delegare allo stessa fase di registrazione delle assenze giornaliera il compito di ampliare di volta in volta la tabella **CodiciAssenze**, sulla base delle reali necessità.

2 - Avvio del programma

Quando si lancia il programma **Adsenze**, appare, dopo qualche secondo, la seguente schermata:



Come si può notare, l'*interfaccia-utente* di **Adsenze** è alquanto diversa da quella di **Anaper**: la barra di *menù* è molto ricca e inoltre lo schermo è suddiviso in partenza in 4 finestre di dimensioni prefissate e invariabili: questa suddivisione indica che lo schermo è già predisposto per la visualizzazione dei Prospetti giornalieri (e dei prospetti di riepilogo, che hanno la stessa struttura) e permette di far scorrere le informazioni mantenendo sempre visibili le intestazioni delle righe (dipendenti) e delle colonne (descrizioni delle assenze); infatti le barre di scorrimento in basso a sinistra e in alto a destra sono inattive (come si può verificare tentando di fare scorrere il cursore dentro la barra) e l'unica finestra che si può *scrollare* in entrambe le dimensioni e quella in basso a destra, destinata a contenere i dati.

Tutte le operazioni di **Adsenze** sono attivate dai comandi di *menù* e quindi ci sembra opportuno procedere nella descrizione del programma illustrando separatamente le funzioni svolte da ogni comando.

3 - Attivazione di un "Prospetto"

Il comando **Prospetto** apre un *menù di pop-up* che contiene le voci **Attiva** e **Disattiva**. Se l'operatore vuole creare (o modificare) un Prospetto, deve cliccare sulla voce **Attiva** (o premere il *tasto acceleratore Home*): questa azione fa comparire una *dialog-box* che chiede di inserire la data del Prospetto da attivare (*default*: la data corrente). La data introdotta è valida se rispetta le regole generali (vedere pag.12) e le regole specifiche di questo caso, che sono:

- la data non deve corrispondere a un giorno "festivo" (vedere pag. 16)
- la data non deve essere posteriore al 31/12 dell'anno corrente (si possono creare o correggere Prospetti "futuri", ma il "futuro" non può andare oltre l'anno in corso);

- se esistono nel *database* Prospetti (anche solo *temporanei*, come quelli creati dalle “previsioni”) con data nel secondo semestre dell’anno corrente, la data non deve essere anteriore al 1/1 dello stesso anno;

- in generale la data non deve essere anteriore all’1/9 dell’anno precedente;

in pratica si possono correggere Prospetti dell’anno precedente (comunque non anteriori all’1/9) solo se, nella creazione dei Prospetti dell’anno in corso, non si è superato il primo semestre. Queste restrizioni sono state introdotte per fare in modo che i semestri con informazioni ancora modificabili siano sempre al massimo 2, e che i dati precedenti siano archiviati come “storici” (è possibile visualizzarli ma non modificarli).

Se la data introdotta è valida e confermata con OK, il programma “lavora” per pochi secondi, durante i quali cerca anzitutto se esiste già un Prospetto con quella data nella tabella **Movimenti** (Prospetto *definitivo*); se non lo trova, lo cerca nella tabella *Provvisori* (Prospetto *temporaneo*); se non lo trova neppure lì, crea un nuovo Prospetto *temporaneo* vuoto. In ogni caso prepara la tabella dei dati da trasferire sulla schermo, che distribuisce nel seguente modo:

- nella finestra in alto a sinistra (che è fissa) scrive la data e il tipo (*definitivo* o *temporaneo*) del Prospetto;
- nella finestra in basso a sinistra (che può scorrere solo verticalmente) introduce le righe con cognome e nome dei dipendenti, estraendo le informazioni dai *records* della tabella **Dipendenti**, ordinati secondo il valore del *campo Nord*;
- nella finestra in alto a destra (che può scorrere solo orizzontalmente) introduce le colonne con descrizione e codice delle assenze (la prima colonna è sempre riservata alla “presenza”); per fare questo il programma estrae le informazioni dai *records* della tabella **CodiciAssenze**, ordinati, da sinistra a destra, secondo i valori decrescenti del *campo Preced* (la “presenza” è in prima colonna perché ha **Preced=99**); non tutti i *records* della tabella, però, vengono trasferiti sullo schermo, ma solo quelli per cui **Preced** è maggiore di zero e, fra agli altri, quelli eventualmente già presenti nel Prospetto; in compenso, viene creata un’ultima colonna con descrizione “ALTRO” che servirà, come vedremo, a “ripescare” dalla tabella i *records* che servono (o a crearne dei nuovi). Questa organizzazione (l’obiettivo è sempre quello di complicare la vita ai programmatori per risparmiarla agli utenti !) ha lo scopo di non creare in partenza la visione di Prospetti inutilmente grossi e difficili da manovrare, ma di presentare ogni volta sullo schermo solo le assenze che capitano più spesso (definite dal fatto che **Preced** ha un valore maggiore di zero) e di lasciare all’operatore il compito di andarsi a prendere solo quelle di cui ha bisogno. La prima volta che si crea un Prospetto compariranno solo tre colonne (“PRESENZA”, “FERIE” e “ALTRO”).
- nella finestra in basso a destra scrive le *caselle* di intersezione fra righe e colonne, che serviranno, come vedremo, a introdurre le informazioni.

Come esempio, prendiamo un database “immaginario”, in cui la tabella **Dipendenti** è quella dell’IRA e le altre tabelle sono quelle di **Proto.mdb**. La seguente figura mostra come appare il primo Prospetto creato da **Adsenze** (ovviamente se ne vede solo una parte):

Adsenze			
Prospetto	Previdori	Visualizza e Stampa	Tabella Codici Assente Erone
PROSPETTO			
TEMPORANEO			
19/02/98			
	PRESENZA (00)	FERIE (02)	ALTRO ()
AMBROSINI ROBERTO			
ARBIZZANI BRUNELLA			
BERGAMINI ROBERTO			
BONDI MARCO			
BRAND JAN			
CESARI PAOLA			
DALLACASA DANIELE			
FERETTI LUCINA			
FICARRA ANTONINO			
GALLERANI ALFREDO			
GIOIA ISABELLA MARIA			
LARI CARLO			
MANFONANI FRANCO			

Notiamo anzitutto che il fondo è diventato azzurro e che è completamente sparito il *system menù* (in alto a destra): la temporanea mancanza del *system menù* impedisce l'interruzione del programma (e anche la sua *iconizzazione*) finché il Prospetto è attivo.

Un'altra osservazione di tipo generale: quando un Prospetto è attivo, tutti gli altri comandi di *menù* sono inefficaci (salvo il comando di disattivazione del Prospetto stesso).

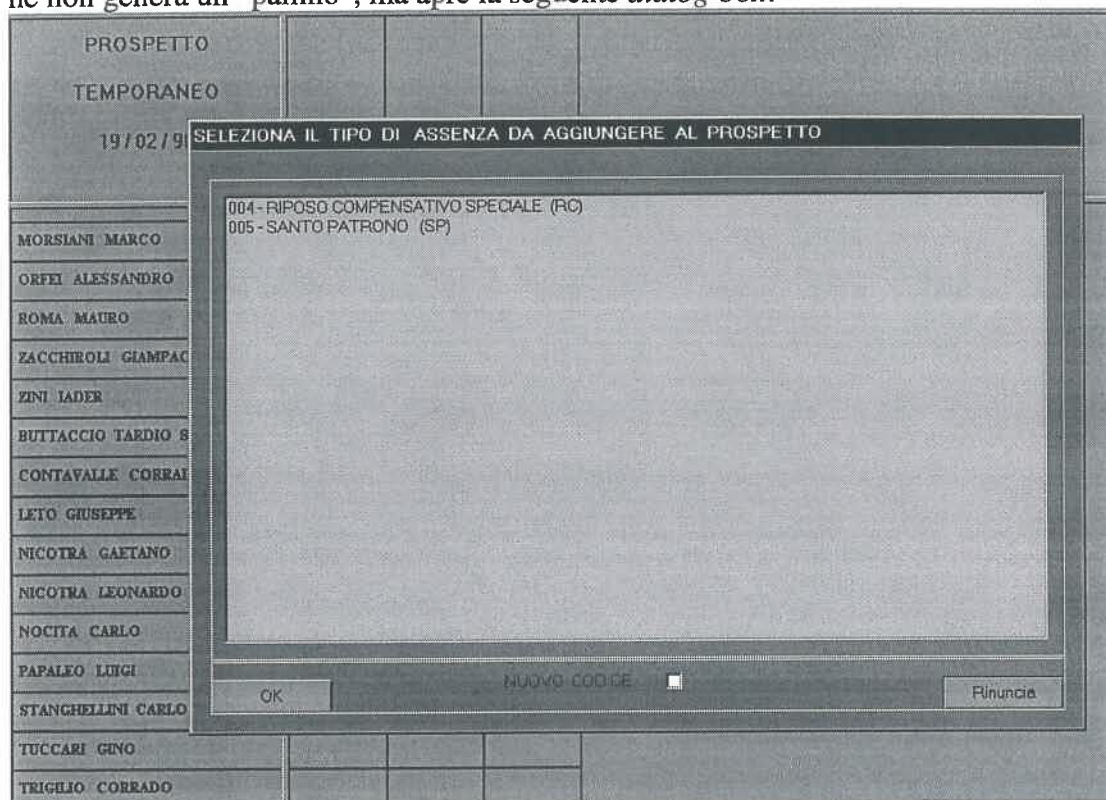
E ora torniamo al nostro esempio: la data è il 19 febbraio (San Corrado, santo patrono di Noto), per cui il giorno è lavorativo per i dipendenti di Bologna e della stazione di Medicina, ma non per quelli della stazione di Noto. L'operatore comincia a inserire i dati, trasferendoli dal materiale cartaceo a sua disposizione: per fare questo deve semplicemente individuare in ogni riga (dipendente) la casella appropriata e poi cliccare: appare un cerchio nero (che d'ora in poi chiameremo "pallino"), che indica, per ogni dipendente, la presenza o il tipo di assenza di quel dipendente in quel giorno. L'inserimento dei "pallini" è un'operazione rapidissima e quasi "divertente": l'unico problema è che l'ordine dei dipendenti sul materiale cartaceo (normalmente secondo l'ora di entrata) e quello sullo schermo di solito non coincidono (per questo è conveniente che i dipendenti siano raggruppati per sede): l'operatore sceglierà a suo piacimento il modo più comodo di procedere.

Tornando all'operazione specifica, precisiamo che:

- non possono esistere più "pallini" sulla stessa riga (ovviamente: non si può essere contemporaneamente assenti per due motivi diversi!): quando si inserisce un "pallino" in una *casella*, quello eventualmente presente in un'altra *casella* della stessa riga scompare automaticamente;
- per cancellare un "pallino" da una *casella* senza (per il momento) sostituirlo con un altro nella stessa riga, bisogna *cliccare* nella *casella* con il tasto destra del *mouse*.

4 - Inserimento di nuove colonne nel Prospetto

Seguendo l'esempio che abbiamo introdotto, l'operatore deve ora inserire l'assenza per "Santo Patrono" dei dipendenti di Noto; questo tipo di assenza non è visualizzato automaticamente in quanto ha **Preced=0** (ovviamente, visto che capita una volta all'anno!). Per cui bisogna ricorrere alla colonna "ALTRO": l'operatore *scrolla* la lista fino in fondo (dove si trovano i dipendenti di Noto), sceglie un dipendente qualsiasi (per esempio il primo) e poi *clicca* sulla *casella* dell'ultima colonna; questa operazione non genera un "pallino", ma apre la seguente *dialog-box*:



la lista che compare nella *dialog-box* (che all'inizio è molto scarna, ma è destinata a crescere col tempo) elenca tutte assenze esistenti nel *database* ma non inserite nel Prospetto; l'operatore seleziona la riga con "Santo Patrono" e poi *clicca* su OK (se *clicca* su "Rinuncia" o preme il tasto **Enter**, annulla l'operazione); ricompare il Prospetto con una colonna in più (inserita prima della colonna "ALTRO" che resta in ultima posizione) e con il "pallino" già al posto giusto. Per quello che riguarda gli altri dipendenti di Noto, non occorre più *clicare* nella colonna "ALTRO", in quanto l'assenza con codice **SP** è già presente nel Prospetto: basterà inserire i "pallini" nel modo normale.

Supponiamo ora che il dipendente FICARRA ANTONINO sia assente per CONGEDO STRAORDINARIO PER VISITA MEDICA, codice **63** (le esatte descrizioni, con relativi codici, dei tipi di assenza sono elencate in tabelle distribuite dal CNR). In questo caso bisogna creare un nuovo *record* nel *database*; l'operatore deve ancora *clicare* nella *casella* di intersezione fra la riga del dipendente e la colonna "ALTRO", ma, quando appare la *dialog-box*, deve selezionare la *checkbox* centrale in basso (vicino alla scritta NUOVO CODICE); fatto questo, la *dialog-box* cambia completamente aspetto e presenta, al posto della lista delle assenze esistenti, una finestra con 4 *caselle*

di testo (corrispondenti ai 4 campi modificabili della tabella **CodiciAssenze**), in cui l'operatore deve inserire i valori del nuovo record, come appare nella figura seguente:

PROSPETTO				
TEMPORANEO				
19/02/98	PRESENZA (00)	FERIE (FE)	SANTO PATRONE (SP)	ALTRO (O)

	AGGIUNGI UN NUOVO RECORD ALLA TABELLA "CODICI ASSENZE"
AMBROSINI ROBERTO	<div style="border: 1px solid black; padding: 5px;"> <p>Descrizione: <input type="text"/></p> <p>Codice: <input type="text"/> Precedenza: <input type="text" value="0"/></p> <p>Descrizione con sillabazione:</p> <input type="text"/> <p>OK <input type="checkbox"/> Nuovo codice <input checked="" type="checkbox"/> Rinuncia</p> </div>
ARBIZZANI BRUNELLA	
BERGAMINI ROBERTO	
BONDI MARCO	
BRAND JAN	
CESARI PAOLA	
DALLACASA DANIELE	
FERETTI LUGINA	
FICARRA ANTONINO	
GALLERANI ALFREDO	
GIOIA ISABELLA MARIA	
LARI CARLO	
MANTOVANI FRANCO	
MINARELLI GIORGIO	
MORGANTI RAFFAELLA	
NANNI MAURO	

Per prima cosa l'operatore deve inserire la frase: **CONGEDO STRAORDINARIO PER VISITA MEDICA** nella casella "Descrizione"; notiamo che, mentre ciò accade, si riempie contemporaneamente con la stessa frase anche la casella "Descrizione con sillabazione": ciò per evitare l'inutile fatica di scrivere la stessa frase due volte e far sì che nella seconda casella l'operatore debba soltanto inserire il carattere "/" per la separazione delle parole in sillabe; completata la prima casella, l'operatore si sposta sulla successiva (può farlo premendo il tasto **TAB**) per inserire il Codice e poi (ancora con **TAB**) sulla successiva per separare le sillabe; infine (sempre con **TAB**) raggiunge la casella "Precedenza", il cui valore di *default* è **0** e non necessita quasi mai di essere modificato (salvo che per certi tipi di assenza abbastanza frequenti, come per esempio le assenze per "missione" o per "malattia", per le quali si ritiene preferibile che la visualizzazione nei Prospetti sia automatica). La creazione del nuovo record nella tabella **CodiciAssenze** avviene se e quando l'operatore conferma *clickando* su **OK**; prima però il programma effettua alcuni controlli e se trova i dati non validi, notifica l'errore impedendo l'uscita dalla *dialog-box*; questi controlli verificano il rispetto delle seguenti regole:

- nessun campo deve essere lasciato in bianco;
- il codice non deve essere il duplicato di un altro codice esistente nella tabella;
- la precedenza deve essere un numero compreso fra **0** e **99**

Se il record è stato creato con successo, si verifica nel Prospetto la trasformazione già vista nel caso di caricamento di un'assenza esistente. Notare la seguente figura:

PROSPETTO TEMPORANEO 19/02/98	PRESENZA (00)	FERIE (FE)	SANTO PA- TRONO (SP)	CONGEDO STRAORDI- NARIO PER VISITA ME- DICA (63)	ALTRO ()
AMBROSINI ROBERTO	●				
ARBIZZANI BRUNELLA	●				
BERGAMINI ROBERTO	●				
BONDI MARCO		●			
BRAND JAN	●				
CESARI PAOLA	●				
DALLACASA DANIELE	●				
FERETTI LUIGINA		●			
FICARRA ANTONINO				●	
GALLERANI ALFREDO	●				
GIOIA ISABELLA MARIA	●				
LARI CARLO	●				
MANTOVANI FRANCO	●				
MINARELLI GIORGIO	●				
MORGANTI RAFFAELLA	●				

nell'intestazione della nuova colonna, la frase di descrizione dell'assenza è spezzata secondo la sillabazione che abbiamo introdotto; ciò chiarisce finalmente a che cosa serve questo *campo* in più: serve a soddisfare esigenze puramente estetiche, cioè a far sì che frasi costrette in uno spazio molto angusto vengano spezzate in modo non troppo sgradevole.

5 - Disattivazione di un "Prospetto"

Quando l'operatore decide che è ora di abbandonare il Prospetto, *clicca* sulla voce **Disattiva** del menù **Prospetto** (oppure preme il *tasto acceleratore End*); il programma chiede conferma (in questo caso **Enter** significa NO) e, se ha risposta positiva, si "mette a lavorare"; infatti è in questa fase che il programma controlla se il Prospetto deve essere *definitivo* o *temporaneo* (indipendentemente da come era al momento dell'attivazione) e, di conseguenza, in quale tabella del *database* devono essere trasferiti i dati (vedere pagg. 15-16). Durante questa fase, il programma può mandare dei messaggi all'operatore, il che avviene per le seguenti informazioni:

- elenco dei dipendenti che non risultano registrati nel Prospetto (questo messaggio non viene inviato se la data del Prospetto è "futura", in quanto i Prospetti "futuri" e perciò inevitabilmente *temporanei*, possono essere "di diritto" incompleti);

- elenco dei dipendenti che hanno esaurito i giorni di ferie spettanti (anche nel “futuro”);
- notifica della avvenuta trasformazione di un Prospetto da *temporaneo* a *definitivo* (o viceversa)

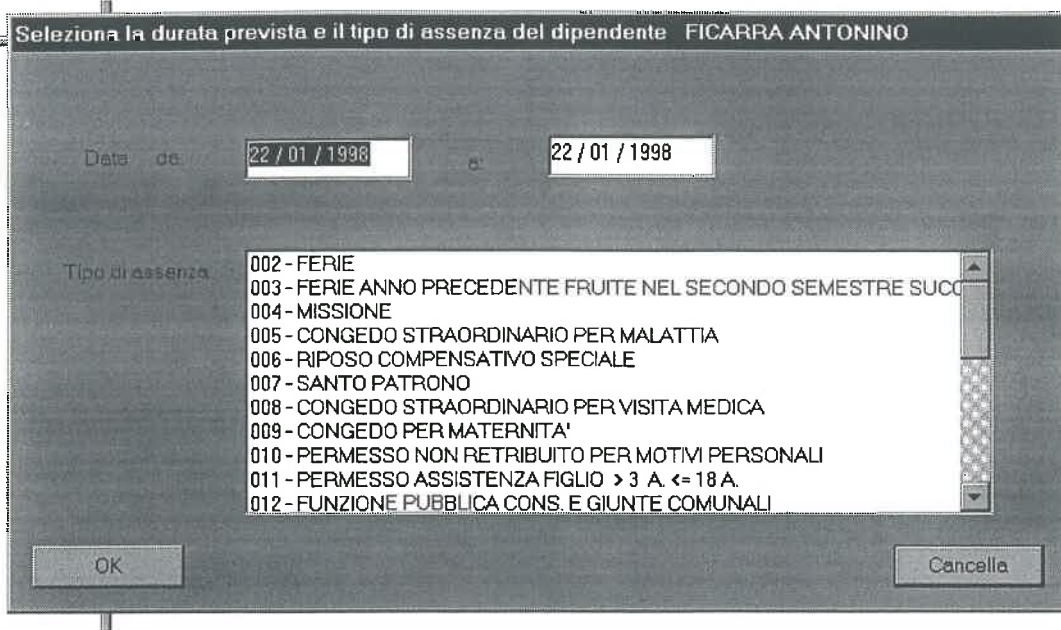
Terminati i controlli il programma cancella il Prospetto dallo schermo (che riassume la colorazione di fondo bianca) e ripristina il *system menù*: da questo momento il programma può di nuovo essere terminato o *iconizzato* e inoltre tornano ad essere attivi gli altri comandi di *menù*.

Un’ultima precisazione prima di terminare questo argomento: il nostro esempio riguardava il caso di un Prospetto nuovo; se invece si attiva un Prospetto già esistente, questo compare sullo schermo con le informazioni che ha, che traduce in “pallini” e sistema nelle *caselle* opportune; nel caso che il Prospetto sia attivato come *temporaneo*, i “pallini” presenti hanno all’inizio colorazione grigia (salvo quelli della colonna delle “presenze”, che sono sempre neri), per ricordare all’operatore che l’informazione è ancora provvisoria (comunque non c’è bisogno di conferma: se il Prospetto diventa *definitivo* e sarà di nuovo attivato, i “pallini” appariranno neri).

6 - Inserimento di “previsioni” di assenza

Il comando **Previsioni** non apre un *menù* di *pop-up*, ma attiva direttamente l’operazione. A differenza del comando **Prospetto**, che crea (o modifica) le informazioni di presenza/assenza di tutti i dipendenti in una sola giornata, il comando **Previsioni** crea (o modifica) le informazioni di assenza di un solo dipendente in una serie di giornate consecutive, e si applica nel caso che il dipendente abbia annunciato in anticipo in quale periodo di tempo intende rimanere assente, consentendo così un notevole risparmio di lavoro per l’operatore e minori possibilità di errore. Le “previsioni di assenza” agiscono esclusivamente su Prospetti *temporanei*, che possono essere creati ex-novo, o anche modificati, nel caso che sullo stesso dipendente e nello stesso periodo di tempo esistano già informazioni registrate precedentemente (che vengono sostituite con le nuove).

Quando l’operatore *clicca* sul comando **Previsioni**, il programma fa anzitutto apparire una *dialog-box*, che contiene l’elenco dei dipendenti, esclusi quelli non più in servizio in data corrente (tralasciamo la figura, per questa volta). Se l’operatore seleziona un dipendente e poi conferma, appare una nuova *dialog-box*, che richiede di selezionare una assenza e definire il periodo di tempo, fornendo le date iniziale e finale. La seguente figura mostra un esempio in cui l’elenco delle assenze é tratto dall’attuale *database* dell’IRA:



Per quello che riguarda l'inserimento delle date iniziale e finale (di *default* viene presentata la data corrente), queste sono soggette, oltre che alle solite regole (vedere pag. 12), alle seguenti restrizioni:

- la data iniziale deve essere posteriore a quella dell'ultimo Prospetto *definitivo* e compresa entro l'anno corrente;
- la data finale deve essere non anteriore alla data iniziale (of course) e compresa entro l'anno corrente (vale sempre la regola che non si possono creare Prospetti dell'anno successivo)

L'elenco dei tipi di assenza comprende tutte le assenze al momento presenti nel *database*. Se la previsione deve riguardare un'assenza non ancora inserita, l'operazione non é fattibile immediatamente: bisogna prima attivare un Prospetto con la data iniziale del periodo e creare la nuova assenza (vedere pagg. 33-34); fatto questo, l'inserimento delle previsioni per il periodo rimanente diventa possibile.

Dopo che l'operatore ha inserito le date, selezionato l'assenza e confermato, il programma "lavora" per un po' e alla fine comunica, con un messaggio, il numero di Prospetti *temporanei* creati e il numero di quelli modificati (i giorni "festivi" sono saltati automaticamente). Attivando successivamente un Prospetto con data compresa nell'intervallo definito, si troverà un "pallino" grigio nella *casella* che identifica il dipendente e l'assenza selezionati nella previsione.

Nel caso che l'assenza selezionata sia di tipo FERIE o FERIE FRUITE NEL SECONDO ... (corrispondenti ai codici FE o 37), il programma esegue, per tutte le giornate del periodo, gli stessi controlli che effettua al momento della disattivazione di un Prospetto giornaliero; se il dipendente raggiunge, a una certa data, il tetto massimo del numero di gg. di ferie spettanti, il programma notifica questo evento con un messaggio.

7 - Visualizzazione e stampa delle tabelle di riepilogo

Il comando **Visualizza e stampa** apre un *menù* di *pop-up* che contiene le voci **Rapporto generico**, **Termina visualizzazione** e **Rapporto mensile**.

Il programma consente di visualizzare e/o stampare tabelle di riepilogo delle presenze/assenze dei dipendenti in un definito periodo di tempo. Le operazioni riguardano esclusivamente i Prospetti *definitivi* e non vengono eseguite se nel periodo selezionato esiste anche una sola giornata (a parte i giorni “festivi”) senza Prospetto o con Prospetto ancora *temporaneo*; in tal caso il programma invia un messaggio di errore con l’elenco delle giornate mancanti. Questa restrizione é necessaria, come misura di controllo della completezza dei dati: può infatti capitare che per dimenticanza un Prospetto resti incompleto e di ciò ci si può accorgere non appena si tenta di costruire un riepilogo. Vedremo più avanti che il programma **Dipcheck** permette invece di visualizzare le informazioni estratte sia dai Prospetti *definitivi* che da quelli *temporanei*.

Se l’operatore vuole conoscere il numero di giorni di presenza/assenza di ogni dipendente, suddivisi per tipo di assenza e calcolati per un certo periodo di tempo, deve selezionare la voce di *menù* **Rapporto generico**: questa operazione attiva una *dialog-box* del tutto identica, nell’apparenza, a quella utilizzata per le “previsioni di assenza”, ma diversa nel significato, in quanto:

- le date iniziale e finale devono definire un periodo di tempo nel “passato” e non nel “futuro”; in particolare il periodo deve essere compreso fra le date del primo e dell’ultimo Prospetto *definitivo* presenti nel *database*;
- nell’elenco delle “assenze” é inclusa anche la PRESENZA;
- si possono selezionare più tipi di assenza contemporaneamente (al limite anche tutti); l’operazione, nota in ambiente **Windows** come *selezione multipla*, si effettua con le seguenti regole (ricordiamole a chi non ha mai avuto occasione di utilizzarle):
 - se si *clicca* su una riga e poi si trascina il *mouse* verso l’alto o verso il basso, vengono selezionate tutte le righe investite dal trascinamento;
 - se si *clicca* su una riga tenendo premuto il tasto *ctrl*, la nuova riga selezionata si aggiunge alle precedenti, che restano selezionate;
 - se si *clicca* su una riga tenendo premuto il tasto *shift*, vengono selezionate tutte le righe comprese fra l’ultima precedentemente selezionata e la nuova riga;
 - se infine si *clicca* su una riga senza premere alcun tasto, si effettua una normale selezione *singola*, cioè le precedenti selezioni vengono rimosse.

Quando l’operatore ha definito l’intervallo di tempo, selezionato le assenze che vuole visionare, e confermato *cliccando* su OK (in questo caso il tasto **Enter** vuol dire “Cancella”), il programma “lavora” per un certo numero di secondi (che dipende dalla lunghezza del periodo e dal numero di assenze selezionate) e poi presenta su video la tabella di riepilogo, che appare come segue (l’esempio riguarda un’operazione effettuata sul *database* dell’IRA):

Adsenze							
Prospetto Emissioni Visualizza e Stampa Tabella Codici Assenze Frame							
SITUAZIONE							
dal 01/01/98							
al 21/01/98							
	PRESENZE / TOTALE	FERIE AN- NO PRECE- DENTE (01)	FESTIVITA' SOTTRES- SE EX LEG. GEN. 97/77 (04)	MISSIONE (02)	CONGEDO STRAORDI- NARIO PER MALATTIA (11-1A)	PERMESSO ASSISTEN- ZA FIGLIO > 3 A. <= 18 A. (17)	DONAZIO- NE SANGUE (07)
ALBERTAZZI VITTORIO	10/13	2	0	0	0	0	0
AMBROSONI ROBERTO	9/13	4	0	0	0	0	0
ARIZZANI BRUNELLA	13/13	0	0	0	0	0	0
BERGAMINI ROBERTO	13/13	0	0	0	0	0	0
BONDI MARCO	11/13	2	0	0	0	0	0
BORTOLOTTI CLAUDIO	10/13	3	0	0	0	0	0
BRAND JAN	12/13	1	0	0	0	0	0
BUTTACCIO TARDIO SALVATORE	8/13	2	0	0	0	0	0
CATTANI ALESSANDRO	11/13	2	0	0	0	0	0
CENARI PAOLA	8/13	2	0	0	0	3	0
CONTAVALLE CORRADO	10/13	3	0	0	0	0	0
DALLACASA DANIELE	11/13	2	0	0	0	0	0
FERETTI LIRGINA	11/13	2	0	0	0	0	0
FICABRA ANTONINO	13/13	0	0	0	0	0	0
GALLERANI ALFREDO	11/13	2	0	0	0	0	0

La figura sembra simile a quella di un Prospetto giornaliero; in realtà le differenze sono molte:

- i dati si riferiscono non a una sola giornata, ma un periodo di più giornate consecutive (infatti nel riquadro in alto a sinistra si leggono due date);
- un Prospetto giornaliero è *interattivo* (cioè l'operatore può modificarlo); in questo caso invece la tabella è di pura visualizzazione (l'unica operazione che si può fare è lo *scrolling*);
- in ogni *casella* non compare un "pallino", ma un numero, che indica per quanti giorni quel dipendente (riga della *casella*) è stato assente per quel tipo di assenza (colonna della *casella*); la prima colonna mostra il numero di giorni di presenza, separato con una barra da un secondo numero, che indica il totale delle giornate lavorative nel periodo di tempo esaminato (questo numero è quasi sempre uguale per tutti; se però un gruppo di dipendenti fosse stato assente per SP, per quei dipendenti il numero di giorni lavorativi sarebbe ridotto di un'unità);
- i dipendenti sono in ordine alfabetico;
- le colonne con tutti zeri non vengono incluse nella tabella (come se le corrispondenti assenze non fossero state selezionate);
- come preannunciato, il programma visualizza i dati delle ferie già ripartiti nei tre codici CNR (nell'esempio la colonna con codice 32 non appare perché avrebbe tutti zeri, in quanto nessuno ha fruito, nel periodo, di giorni di "ferie dell'anno corrente").

Quando l'operatore ritiene di avere esaminato a sufficienza la tabella di riepilogo, clicca sulla voce di menù **Termina visualizzazione** del comando **Visualizza e stampa** (oppure preme il tasto *acceleratore* **End**); il programma, prima di rimuovere la tabella dello schermo, chiede se deve creare un file di stampa e, se l'operatore ha *cliccato* su SI (in questo caso il tasto **Enter** equivale a NO), costruisce il file, il cui contenuto, continuando con il nostro esempio, è mostrato nella figura seguente:

26205721.txt - Blocco note

File Modifica Cerca ?

SITUAZIONE PRESENZE E ASSENZE DEI DIPENDENTI
dal 01/01/1998 al 21/01/1998

ALBERTAZZI VITTORIO.....	10/13	2(31)
AMBROSINI ROBERTO.....	9/13	4(31)
ARBIZZANI BRUNELLA.....	13/13	
BERGAMINI ROBERTO.....	13/13	
BONDI MARCO.....	11/13	2(31)
BORTOLOTTI CLAUDIO.....	10/13	3(31)
BRAND JAN.....	12/13	1(31)
BUTTACCIO TARDIO SALVATORE..	8/13	2(31)
CATTANI ALESSANDRO.....	11/13	2(31)
CESARI PAOLA.....	8/13	2(31) 3(17)
CONTAVALLE CORRADO.....	10/13	3(31)
DALLACASA DANIELE.....	11/13	2(31)
FERETTI LUIGINA.....	11/13	2(31)
FICARRA ANTONINO.....	13/13	
GALLERANI ALFREDO.....	11/13	2(31)
GIOIA ISABELLA MARIA.....	0/13	13(92)
LARI CARLO.....	13/13	
LETO GIUSEPPE.....	11/13	2(31)

Nel file di stampa, le colonne della tabella sono "compattate" (per motivi di spazio): la prima colonna (presenze/totale), resta inalterata, mentre le altre sono trascritte solo se il numero di giorni é diverso da zero; ogni numero é accompagnato fra parentesi dal codice dell'assenza a cui si riferisce (per esempio FERETTI LUIGINA é stata presente 11 giorni su 13 e assente 2 giorni per ferie, codice 31).

8 - Rapporti mensili

La preparazione dei Rapporti mensili, da trascrivere sull'apposito modulo che deve essere inviato al CNR, costituisce uno dei principali obiettivi del programma (vedere pag. 1).

Quando l'operatore *clicka* sulla voce di **menù Rapporto mensile** del comando **Visualizza e stampa**, il programma fa apparire la seguente *dialog-box*:



L'operatore deve selezionare il mese (che si intende di *default* dell'anno corrente; per ottenere che il mese sia dell'anno precedente bisogna selezionare anche la *checkbox* in basso); quando l'operatore conferma con OK (o con **Enter**), il programma parte con la creazione di un file di stampa (questa operazione non ha un output su video).

Nella figura mostriamo un esempio di Rapporto mensile (tratto, come al solito, dal *database* dell'IRA):

The screenshot shows a Notepad window titled "DICEMBRE97.txt - Blocco note". The window displays a report titled "SITUAZIONE PRESENZE E ASSENZE DEI DIPENDENTI dal 01/12/1997 al 31/12/1997 - PROSPETTO MENSILE". The report lists names and their attendance/absence records for December 1997.

Nome	01/12	02-08/12	09-15/12	16-19/12	20-24/12	25-31/12
ALBERTAZZI UITTO	11/20	17-19(32)	22-24(32)	29-31(32)		
AMBROSINI ROBERT	18/20	16(92)	18(92)			
ARBIZZANI BRUNEL	17/20	2(32)	15(32)	24(32)		
BERGAMINI ROBERT	16/20	15(32)	22-24(32)			
BONDI MARCO.....	16/20	24(32)	29-31(32)			
BORTOLOTTI CLAUD	9/20	4(92)	5(RC)	16-19(32)	22-24(32)	29-31(32)
BRAND JAN.....	14/20	22-24(32)	29-31(32)			
BUTTACCIO TARDIO	14/20	5(32)	15-16(RC)	22-24(32)	30-31(32)	
CATTANI ALESSAND	12/20	11(32)	18(32)	22-24(32)	29-31(32)	
CESARI PAOLA....	14/20	3(17)	17(17)	24(17)	29-31(32)	
CONTAVALLE CORRA	20/20					
DALLACASA DANIEL	19/20	24(32)				
FERETTI LUIGINA.	18/20	24(32)	31(32)			
FICARRA ANTONINO	17/20	29-31(32)				
GALLERANI ALFRED	20/20					
GIOIA ISABELLA M	0/20	1-5(92)	9-12(92)	15-19(92)	22-23(92)	24(94) 29-31(32)
LARI CARLO.....	17/20	22-24(32)				
LETO GIUSEPPE...	12/20	2-5(92)	9(92)	11-12(32)	31(32)	

Il Rapporto contiene le informazioni che vuole il CNR. Notiamo che:

- i nomi sono in ordine alfabetico (abbreviati per ragioni di spazio);
- la prima colonna dopo i nomi ha lo stesso significato (giorni di presenza / totale dei giorni lavorativi) che aveva nelle tabelle di riepilogo;
- di seguito sono indicate le date (non il numero, come nelle tabelle di riepilogo) dei giorni di assenza con, fra parentesi, il codice del tipo di assenza;
- quando si ha una stessa assenza per più giorni consecutivi, vengono indicate le date del primo e dell'ultimo giorno, separate da un trattino;
- le assenze per **RC** e **SP** servono per uso interno e non vanno (ovviamente) trascritte nel modulo del CNR (notare che il conteggio delle presenze include anche le assenze per **RC**); se vi fosse un'assenza per **SP**, il programma avrebbe ridotto di uno il numero totale di giorni lavorativi sulla stessa riga.

Si raccomanda di cambiare nome ai files dei Rapporti mensili e di conservarli (i nostri li abbiamo chiamati LUGLIO97.txt, AGOSTO97.txt ecc..). Questi files contengono tutta l'informazione essenziale e possono costituire, nel futuro, un archivio "storico".

9 - Gestione della tabella "CodiciAssenze"

Il comando **Tabella Codici Assenze** apre un *menù* di *pop-up* che contiene le voci **Modifica Record**, **Cancella Record** e **Stampa**. Manca la voce **Inserisci Record**, in quanto, come abbiamo visto (pagg. 33-34), la creazione di nuovi *records* avviene in modo "dinamico", durante la registrazione dei Prospetti giornalieri, ogni volta che se ne presenta la necessità.

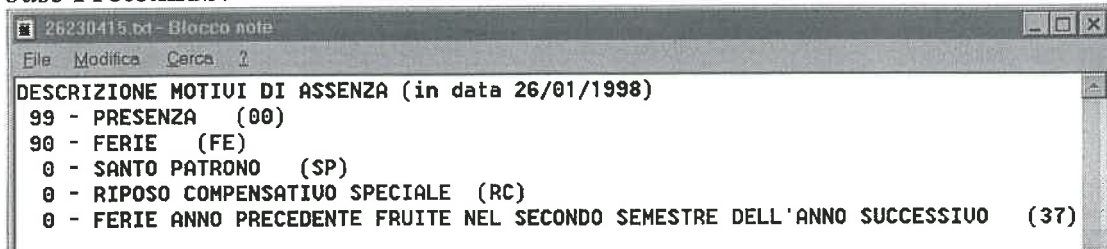
In realtà l'esigenza di modificare o addirittura cancellare *records* nella tabella é quasi inesistente. La modifica potrebbe essere necessaria, per esempio, per correggere errori di ortografia nelle descrizioni, o, più spesso, per cambiare il valore del *campo* **Preced**, nel caso che l'esperienza suggerisca di includere nella (o togliere dalla) visualizzazione automatica certi tipi di assenza che si verificano più (o meno) spesso di altri. Ancora più rara ci sembra la cancellazione (é stato inserito un tipo di assenza che in realtà non serve e lo si vuole "togliere dai piedi").

Quando l'operatore *clicka* sulla voce **Modifica Record** appare una *dialog-box* identica a quella che si presenta quando é selezionata la colonna "ALTRO" dei Prospetti giornalieri (vedere pagg. 33-34); c'è un'unica differenza: la *check-box* in basso non serve per scegliere fra due alternative (la selezione nella lista delle assenze esistenti oppure l'inserimento di una assenza nuova), ma per confermare la selezione di una assenza esistente onde permettere al programma di presentare l'altra forma della *dialog-box*, dove compaiono le 4 *caselle di testo* in cui inserire le modifiche. Nell'effettuare queste operazioni, tenere presente che i dati modificati sono soggetti agli stessi controlli di validità dei dati creati ex-novo (vedere pag. 34).

Quando l'operatore *clicka* sulla voce **Cancella Record** appare ancora la stessa *dialog-box*; questa volta però la *check-box* manca del tutto (infatti l'operazione si esaurisce nella selezione dell'assenza da cancellare e non serve passare alla seconda forma della *dialog-box*). Come per i *records* della tabella **Dipendenti** (vedere pag. 27), il pro-

gramma si rifiuta di cancellare un'assenza se il suo codice é già presente in almeno un Prospetto (prima bisogna rimuoverla dal Prospetto e poi la si può cancellare).

Infine, quando l'operatore *clicca* sulla voce **Stampa**, il programma prepara un file di stampa del contenuto della tabella. La figura seguente mostra il file ricavato dal *database Proto.mdb*:

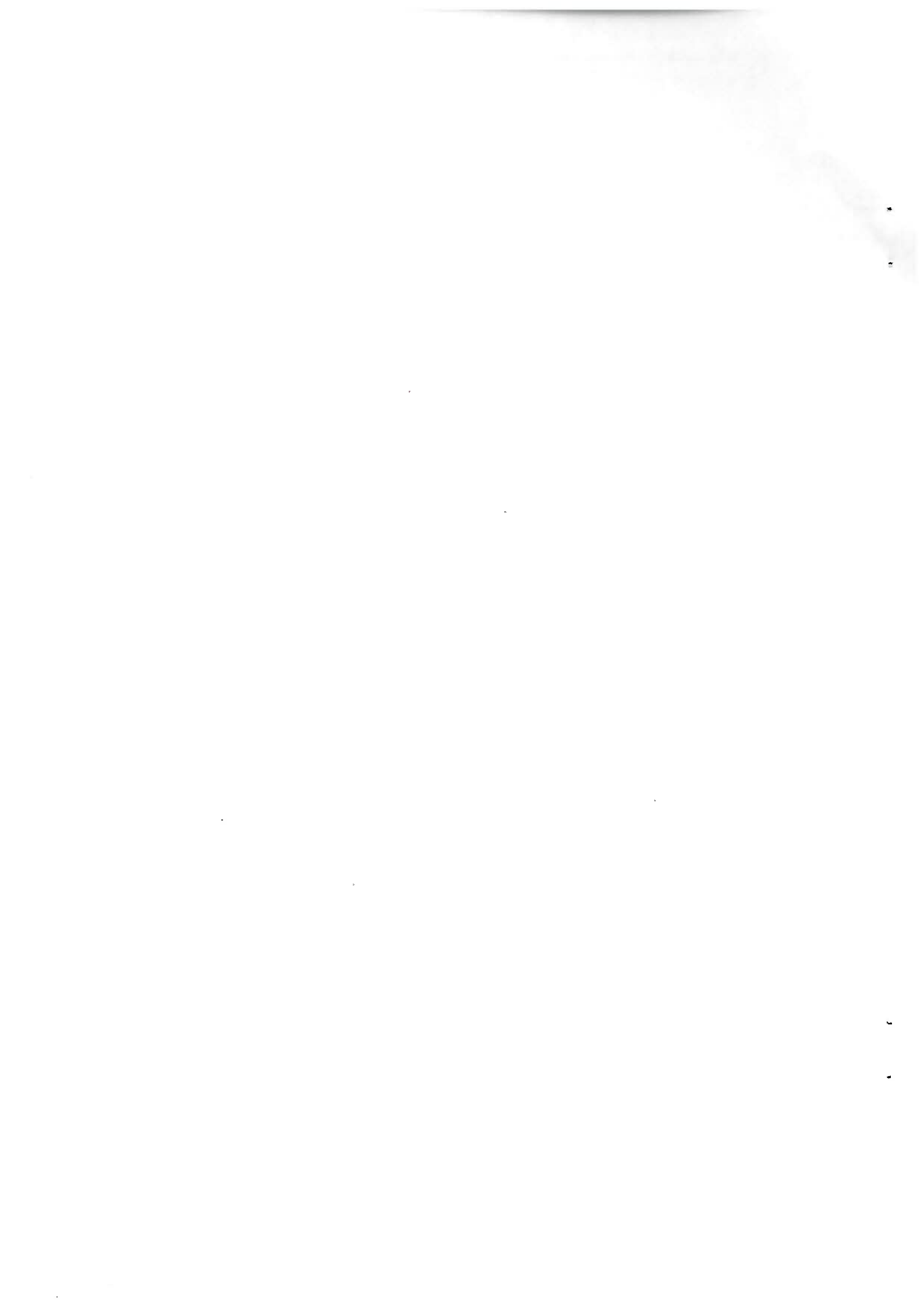


```
26230415.txt - Blocco note
File Modifica Cerca ?
DESCRIZIONE MOTIVI DI ASSENZA (in data 26/01/1998)
99 - PRESENZA (00)
90 - FERIE (FE)
0 - SANTO PATRONO (SP)
0 - RIPOSO COMPENSATIVO SPECIALE (RC)
0 - FERIE ANNO PRECEDENTE FRUITE NEL SECONDO SEMESTRE DELL'ANNO SUCCESSIVO (37)
```

In ogni riga (che corrisponde a un *record*) il primo numero indica il valore di **Preced**, seguono la descrizione e il codice, fra parentesi. Notare che nell'intestazione appare la data di creazione del file: questo perché la tabella **CodiciAssenze** é in continua espansione e quindi é importante sapere quando il file é stato prodotto.

10 - Comando "Frame"

Il comando **Frame** apre un *menù* di *pop-up* che contiene le voci **Barra di stato** e **Icona** (quest'ultima ha soltanto un valore ... "pubblicitario"); per quello che riguarda la *barra di stato* (che si trova in fondo allo schermo), é importante che essa resti visibile, perché ogni volta contiene una breve spiegazione del significato della voce di *menù* selezionata con il posizionamento del *mouse*.



Programma "Dipcheck"

1 - Considerazioni generali

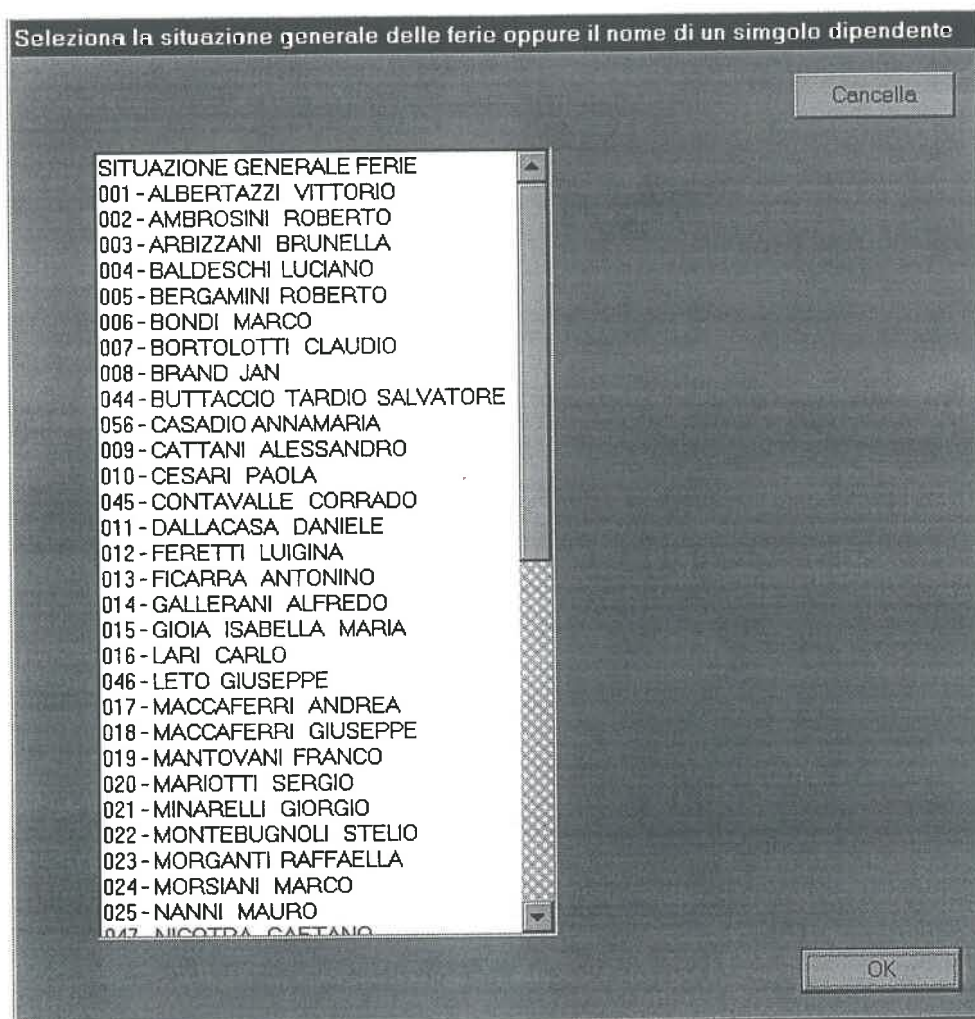
Dipcheck é un programma di pura visualizzazione: accede al *database* in sola lettura e pertanto può essere usato da chiunque (a differenza degli altri 3 programmi, che sono riservati agli "addetti ai lavori"). Anzi, **Dipcheck** é stato fatto proprio a questo scopo (vedere pag.3).

Esiste però un problema, di cui noi stessi non ci siamo accorti immediatamente: quando é in esecuzione un qualunque programma che accede al *driver ODBC*, lo stesso *driver* ha la (cattiva) abitudine di creare dei files temporanei nella stessa directory nel programma e di cancellarli quando il programma é terminato. In una organizzazione di rete ciò ha un effetto "micidiale", in quanto impedisce l'utilizzo dei programmi dagli altri PC, a meno che la directory in cui risiedono i programmi non sia di libero accesso, sia in lettura che in scrittura (cosa che é altamente sconsigliabile). Se i programmi (e il *database*) risiedono in un PC con **Windows NT Server**, il problema si potrebbe risolvere rendendo libero l'accesso alla directory e proteggendo i singoli files; tale soluzione, che comunque é poco soddisfacente, é impraticabile negli Istituti che sono dotati soltanto di PC con **Windows 95** collegati in rete. In questi casi riteniamo che l'unico compromesso possibile fra le due esigenze contrastanti (protezione dei dati e possibilità di operare in rete) sia il seguente:

- i programmi che accedono al *database* in lettura e scrittura possono essere eseguiti nel solo PC in cui risiedono (presumibilmente il PC della segreteria, il cui uso deve essere vietato agli altri dipendenti);
- per consentire l'utilizzo di **Dipcheck** da qualunque utente collegato in rete, occorre creare una directory specifica, di libero accesso in lettura e scrittura e poi copiare il file **Dipcheck.exe** in questa directory; il *database* deve invece risiedere in una directory accessibile solo in lettura; infine gli altri programmi, compreso l'originale **Dipcheck.exe**, devono essere conservati in una directory inaccessibile. Gli utenti eseguiranno l'unico **Dipcheck** visibile dalla rete, e potranno farlo senza problemi in quanto i files temporanei creati da **Dipcheck** vanno a finire in una directory accessibile in scrittura.

2 - Avvio del programma

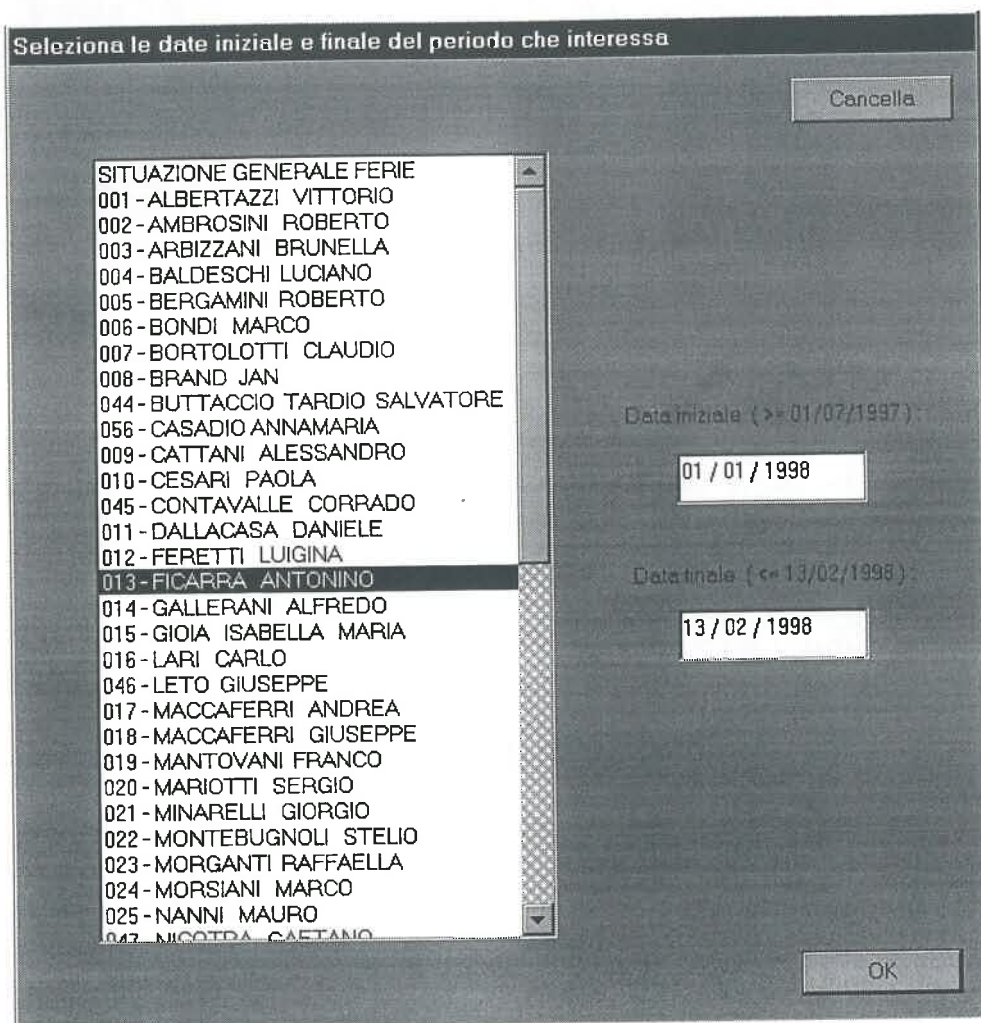
Dipcheck é un programma semplicissimo da usare ed esegue una sola operazione, selezionata non da *menù* (che é inesistente) ma da una *dialog-box*, che appare non appena si lancia il programma. Come vedremo, questa *dialog-box*, che costituisce l'unico punto di interazione fra operatore e programma, é del tutto particolare, in quanto cambia di aspetto in funzione del tipo di selezione effettuata. L'esempio della figura che segue proviene, come al solito, dal *database* dell'IRA:



Come si può notare la *dialog-box* contiene all’inizio soltanto una lista, con l’elenco dei dipendenti in ordine alfabetico (ignorare il numero di codice sulla sinistra di ogni nome, che serve solo al programma). A differenza da altre liste analoghe, però, notiamo c’è una riga in più, la prima, con la scritta “SITUAZIONE GENERALE FERIE”; questa riga, se selezionata, attiverà un tipo diverso di operazione, come vedremo. Per il momento la ignoriamo.

3 - Selezione del nome di un dipendente

Se l’operatore è interessato a conoscere la situazione delle presenze e assenze di un dipendente in un dato periodo di tempo, *clicca* sul nome del dipendente (ci siamo doverosamente informati: non esistono in questo caso vincoli di *privacy*, in quanto le presenze e assenze di un dipendente sono informazioni pubbliche; anche nel caso di assenza per malattia il codice CNR è generico e non chiede di specificare di quale malattia si tratti: se così fosse, sarebbe invece un’informazione riservata); come conseguenza, la *dialog-box* cambia aspetto e introduce due *caselle di testo*, dove l’operatore deve specificare le date iniziale e finale del periodo che gli interessa. Seguendo nel nostro esempio, supponiamo che l’operatore abbia selezionato il dipendente FICARRA ANTONINO:



Notiamo anzitutto che la frase di intestazione della *dialog-box* é cambiata e che sopra entrambe le *caselle di testo* appaiono delle scritte, dove vengono specificati i limiti inferiore e superiore dell'intervallo di tempo da selezionare; in particolare:

- **Data iniziale (>= 1/7/1997)** : significa che la data iniziale non può essere anteriore all' 1/7/97
- **Data finale (<= 13/2/1998)** : significa che la data finale non può essere posteriore al 13/2/98

Questi limiti vengono desunti attingendo le informazioni dalle tabelle **Movimenti e Provvisori** del *database* (**Dipcheck** esamina sia i Prospetti *definitivi* che quelli *temporanei*) e rappresentano le date del primo e dell'ultimo Prospetto in cui si trova il dipendente selezionato (possono cambiare se si cambia la selezione del dipendente). Inoltre il periodo di tempo non può scavalcare un anno (per questo motivo il valore di *default* della data iniziale é 1/1/1998 anziché 1/7/1997); nel caso che si desideri visualizzare informazioni relative sia al 1997 che al 1998 bisogna eseguire **Dipcheck** due volte: la prima volta con data finale 31/12/1997 (e modificando la data iniziale in modo che appartenga al 1997), la seconda volta lasciando il *default* 1/1/1998 della data iniziale. Per il resto le regole per l'inserimento delle date sono quelle solite (vedere pag. 12), con l'ovvia ulteriore condizione che la data iniziale non sia posteriore alla data finale.

Eseguita e confermata la selezione dell'intervallo di tempo (supponiamo, seguendo l'esempio, di non modificare i *defaults*), il compito dell'operatore é praticamente finito: il programma "lavora" per un po', e dopo presenta la seguente schermata:

INFORMAZIONI SUL DIPENDENTE: PICARRA ANTONINO
NEL PERIODO: 01/01/1998 - 13/02/1998

NEL PASSATO:	LUNEDI'	MARTEDI'	MERCOLEDI'	GIOVEDI'	VENERDI'
29/12 - 02/01					Pr
05/01 - 09/01	Pr		Pr	92	92
12/01 - 16/01	Pr	Pr	31	31	31
19/01 - 23/01	Pr	Pr	Pr		

RIEPILOGO (NUMERO DI GIORNI DI ASSENZA RAGGRUPPATI PER CODICE):
 PRESENZA (Pr): 8/13
 FERIE ANNO PRECEDENTE (31): 3
 MISSIONE (92): 2

GIORNI DI FERIE DA FARE: ANNO CORRENTE (IN DATA 21/01/98) = 32
 ANNO PRECEDENTE = 14

IN PREVISIONE:	LUNEDI'	MARTEDI'	MERCOLEDI'	GIOVEDI'	VENERDI'
19/01 - 23/01				31	31
26/01 - 30/01	31	31	31	31	31
02/02 - 06/02	31	31	31	31	31
09/02 - 13/02	31	31	94	94	94

Le informazioni che appaiono su video sono suddivise in quattro gruppi, caratterizzati anche da una diversa colorazione:

- nel primo gruppo (di colore blu) sono indicati i codici delle presenze (in questo caso abbiamo preferito "Pr", che é più chiaro di "00") e delle assenze del dipendente per ogni giornata, disposti in modo che a ogni riga corrisponda una settimana; queste informazioni sono desunte dai Prospetti *definitivi* (la frase "NEL PASSATO:" vuol dire sostanzialmente questo); dove l'assenza é per ferie, il tipo di ferie é desunto dalla tabella **FerieInfo**; infine notare che in corrispondenza dei giorni festivi (per esempio il 6/1) non viene scritto nulla;
- nel secondo gruppo (di colore verde) sono mostrati i dati riepilogativi della tabella precedente, espressi in termini di numero di giorni di un dato tipo di assenza; con l'occasione viene anche spiegato il significato di ciascun codice (il normale utente potrebbe anche non conoscerlo); per quanto riguarda la presenza, il secondo numero dopo la barra indica, al solito, il totale dei giorni lavorativi nel periodo di tempo ricoperto dalla prima tabella;
- il terzo gruppo (di colore viola) é forse quello che contiene l'informazione più interessante: quanti giorni di ferie si possono ancora fare, dell'anno corrente (codici 94 e 32 insieme) e dell'anno precedente (codice 31); questa informazione non tiene conto delle eventuali ferie "in previsione" e, indipendentemente dal periodo di tempo selezionato, si riferisce sempre alla data dell'ultimo Prospetto *definitivo*;
- il quarto gruppo (di colore rosso) contiene gli eventuali dati di assenza "in previsione", estratti dai Prospetti *temporanei*; la tabella é strutturata come quella del primo gruppo; notare, nell'esempio, che il dipendente esaurirà le ferie dell'anno precedente il 10/2 e quindi le ferie successive avranno codice 94 (festività soppresse).

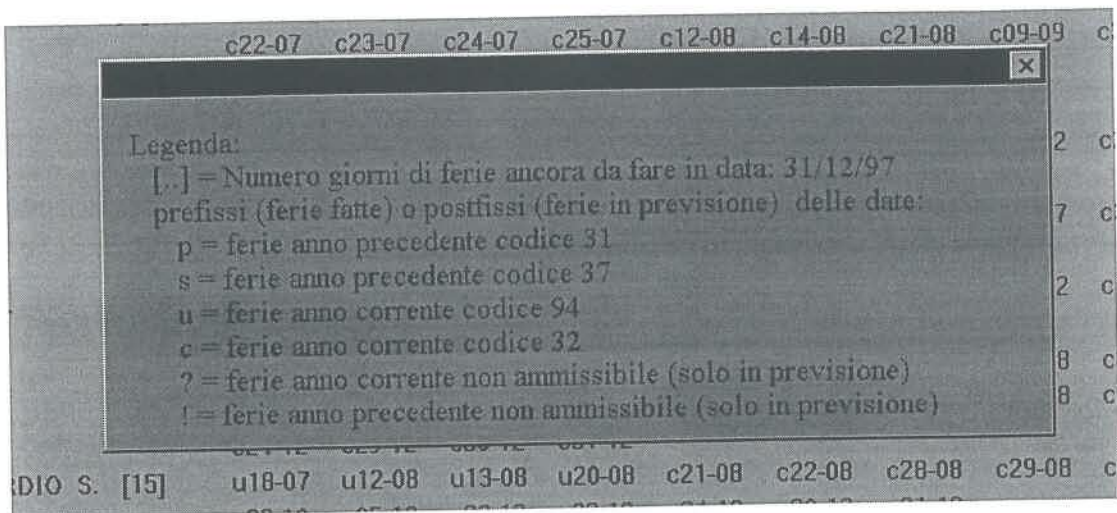
Quando l'operatore ha esaminato con suo comodo i dati sul video, non gli resta da fare che chiudere il programma, *cliccando* sulla **X** del *system menù* ; il programma, prima di uscire, gli chiede se vuole creare il file di stampa (in questo caso il tasto **Enter** significa NO). Non riportiamo immagini del file, in quanto il suo contenuto é strutturato in modo identico ai dati su video.

4 - Selezione della "Situazione generale delle ferie"

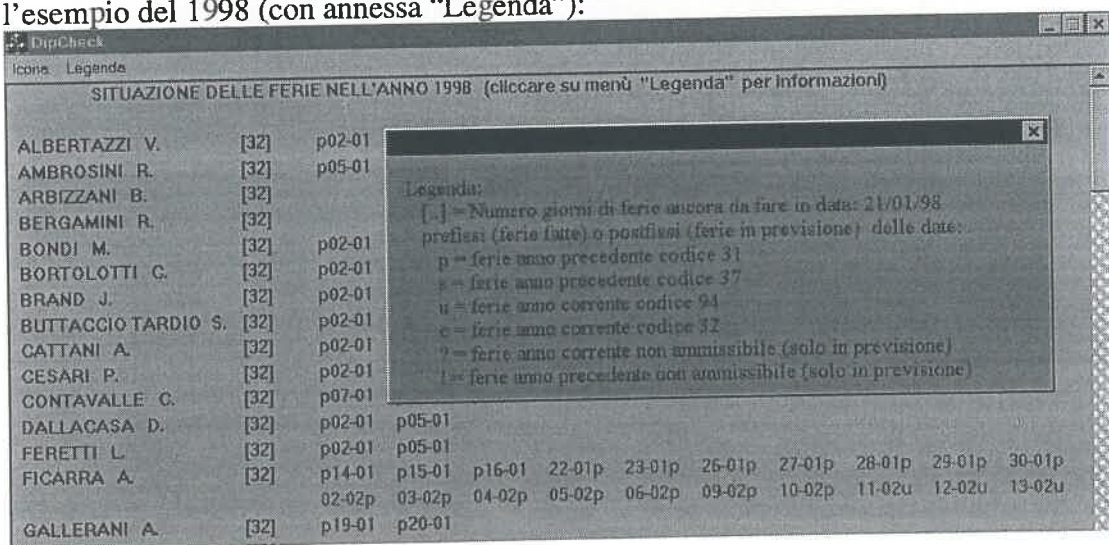
Tornando alla *dialog-box* iniziale, se l'operatore vuole conoscere le giornate di ferie (fatte o in previsione di fare) di tutti i dipendenti in servizio in un certo anno, *clicca* sulla prima riga: nella *dialog-box* appare una *casella di testo* in cui gli viene richiesto di specificare l'anno; eseguita (e confermata) tale operazione, il controllo passa al programma, che, dopo alcuni secondi, mostra la seguente schermata (nell'esempio abbiamo scelto il 1997 perché contiene una maggiore quantità di informazioni, trattandosi, al solito, di dati estratti dal database dell'IRA):

SITUAZIONE DELLE FERIE NELL'ANNO 1997 (cliccare su menù "Legenda" per informazioni)											
ALBERTAZZI V.	[2]	u14-07	u15-07	u21-07	u22-07	c23-07	c24-07	c25-07	c28-07	c29-07	c30-07
		c31-07	c01-08	c11-08	c12-08	c13-08	c14-08	c18-08	c19-08	c20-08	c21-08
		c22-08	c17-12	c18-12	c19-12	c22-12	c23-12	c24-12	c29-12	c30-12	c31-12
AMBROSINI R.	[21]	u17-07	u18-07	u04-08	u05-08	c06-08	c07-08	c08-08	c11-08	c12-08	c13-08
		c14-08									
ARBIZZANI B.	[7]	u08-07	u09-07	u10-07	c11-07	c14-07	c15-07	c16-07	c17-07	c18-07	c21-07
		c22-07	c23-07	c24-07	c25-07	c12-08	c14-08	c21-08	c09-09	c30-09	c16-10
		c23-10	u12-11	c02-12	c15-12	c24-12					
BALDESCHI L.	[18]										
BERGAMINI R.	[21]	u22-07	u12-08	u13-08	u14-08	c16-09	c17-10	c14-11	c15-12	c22-12	c23-12
		c24-12									
BONDI M.	[15]	u18-07	u21-07	u22-07	u23-07	c24-07	c25-07	c28-07	c29-07	c30-07	c31-07
		c01-08	c14-08	c24-11	c24-12	c29-12	c30-12	c31-12			
BORTOLOTTI C.	[15]	u06-08	u07-08	u08-08	u11-08	c12-08	c13-08	c14-08	c16-12	c17-12	c18-12
		c19-12	c22-12	c23-12	c24-12	c29-12	c30-12	c31-12			
BRAND J.	[1]	u16-07	u28-07	u29-07	c30-07	c31-07	c01-08	c04-08	c05-08	c06-08	c07-08
		c08-08	c11-08	c12-08	c13-08	c14-08	c18-08	c19-08	c20-08	c22-12	c23-12
		c24-12	c29-12	c30-12	c31-12						
BUTTACCIO TARDIO S.	[15]	u18-07	u12-08	u13-08	u20-08	c21-08	c22-08	c28-08	c29-08	c02-09	c17-10
		c29-10	c05-12	c22-12	c23-12	c24-12	c30-12	c31-12			
CASADIO A.	[0]	u30-10	c04-11	c11-11	c17-11	c20-11	c24-11	c25-11			
GATTANI A.	[15]	u14-07	u15-07	u16-07	u17-07	c15-09	c16-09	c17-09	c18-09	c19-09	c11-12
		c18-12	c22-12	c23-12	c24-12	c29-12	c30-12	c31-12			
CESARI P.	[20]	u07-08	u08-08	u11-08	u12-08	c13-08	c14-08	c26-08	c12-11	c26-11	c29-12
		c30-12	c31-12								

I dipendenti sono elencati in ordine alfabetico. Tutti i numeri che appaiono (eccetto quelli fra parentesi quadre) sono date (giorno-mese); poiché le informazioni sono molto compatte, e quindi un po' "criptiche", é necessario che vengano fornite delle spiegazioni; a questo scopo si attiva una voce di *menù* (**Legenda**), *cliccando* sulla quale l'operatore ottiene l'apertura di una *dialog-box* che contiene le chiavi di lettura dei simboli che appaiono accanto alle date:



notare che questa volta il numero di giorni di ferie ancora da fare si riferisce all'ultimo giorno, dell'anno selezionato, in cui esistono informazioni nei Prospetti *definitivi* (in questo caso il 31/12, essendo il 1997 già completato). La *dialog-box* può essere rimossa *clickando* sulla sua casella con la X. Dal contenuto della "Legenda" si capisce che le letterine che accompagnano le date indicano il tipo di ferie, e che, se sono davanti alle date (*prefissi*), l'informazione è tratta da Prospetti *definitivi*, mentre, se sono scritte dopo le date (*postfissi*), si tratta di ferie "in previsione", cioè desunte da Prospetti *temporanei*. Nell'esempio mancano casi di *postfissi* in quanto i Prospetti del 1997 sono tutti *definitivi*; perciò facciamo vedere, senza ulteriori commenti, anche l'esempio del 1998 (con annessa "Legenda"):



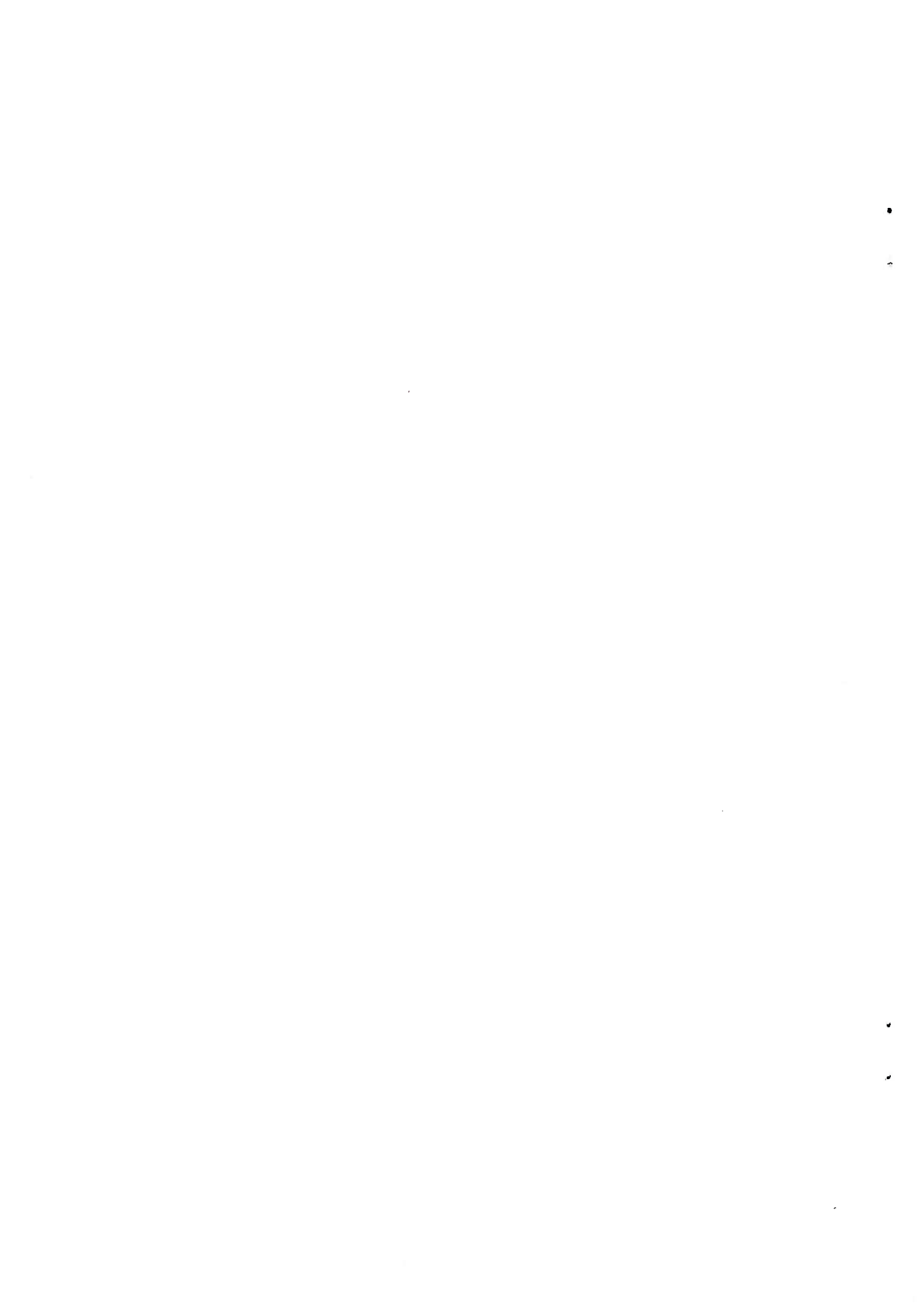
In uscita dal programma, se l'operatore accetta di creare il file di stampa, viene generato un file identico nella struttura ai dati su video, salvo il fatto che le informazioni della "Legenda" vengono scritte insieme agli altri dati (per la precisione, all'inizio del file).

5 - Per gli utenti più “pigri”

Dipcheck é provvisto di un *timer*, che, trascorso un certo periodo di tempo, lo fa terminare automaticamente. L'utente ha a disposizione, nelle varie fasi dell'esecuzione del programma, i seguenti tempi:

- 3 minuti per effettuare la selezione e uscire dalla *dialog-box* iniziale;
- 10 minuti per esaminare (comodamente) i dati;
- 1 minuto per prendere atto del nome del file di stampa (se lo ha creato).

L'introduzione di questo “dispositivo a orologeria” é stata decisa come misura di precauzione nei confronti di utenti particolarmente “pigri” (o distratti): potrebbe capitare (anzi, é già capitato!) che un dipendente, soddisfatta la sua curiosità, si dimentichi di chiudere il programma; ciò impedirebbe successivamente agli addetti di far partire **Adsenze** per la registrazione dei dati giornalieri: infatti, mentre in sola lettura il *database* può essere aperto da più programmi contemporaneamente, un programma che scrive nel *database* deve averne l'accesso esclusivo.



Programma “Transfer”

Il programma **Transfer** serve per riallineare il contenuto della tabella **FerieInfo** a quello delle tabelle **Movimenti** e **Provvisori**, nel caso che i dati di **FerieInfo** non siano (o si sospetta che non siano) più validi. Questo può verificarsi in due circostanze:

- 1) se, per un malaugurato incidente, la tabella si è rovinata (e allora non c'è niente da fare);
- 2) se, con **Anaper**, è stato modificato il numero di giorni di ferie spettanti a un dipendente e, nello stesso anno, esistono già registrazioni di ferie di quel dipendente (vedere pagg. 19-20); in questo caso è lo stesso **Anaper** ad avvisare che potrebbe essere necessario far girare **Transfer**, in quanto il dato che è stato modificato potrebbe influire sulla ripartizione delle ferie nei tre codici (e sul controllo dell'esaurimento delle ferie) e queste informazioni, calcolate di volta in volta da **Adsenze** e memorizzate nella tabella **FerieInfo**, potrebbero necessitare di un aggiornamento.

Transfer è ancora più semplice ed essenziale di **Dipcheck**: quando si lancia il programma, appare la (solita) *dialog-box* con il (solito) elenco dei dipendenti in ordine alfabetico (tralasciamo la figura, che ormai è ben nota). La parte della tabella **FerieInfo** che deve essere ricostruita potrebbe riguardare uno o più dipendenti, o al limite anche tutti, se si vuole ricostruire la tabella per intero. Per questo motivo, il programma prevede la *selezione multipla* (per le regole della *selezione multipla*, vedere pag. 38) di un numero qualsivoglia di dipendenti e poi lavora sequenzialmente su ogni dipendente.

Una volta che l'operatore ha selezionato il dipendente (o i dipendenti) e confermato, ha già esaurito il suo compito: il programma esegue le correzioni e, come unica prova che è “vivo”, mostra nella *barra di stato* (in aggiornamento continuo) il nome del dipendente su cui sta lavorando e il numero di *records* della tabella prima cancellati e poi ricreati. Terminato un dipendente, il programma si arresta per qualche secondo e poi parte con il prossimo dipendente.

Alla fine il programma esce automaticamente.

Se si fa girare **Transfer** quando non è necessario, l'operazione, anche se inutile, non produce alcun danno, e quindi è consigliabile farlo girare anche se c'è solo il minimo sospetto che i dati della tabella non siano più validi (in pratica, seguire sempre il “consiglio” di **Anaper**).



Conclusioni e ultime raccomandazioni

Le procedure di Gestione del Personale sono funzione presso l'IRA da circa sei mesi, con risultati molto soddisfacenti.

I programmi sono il più possibile protetti da errori operativi (ma non mettiamo limiti alla fantasia umana), e non dovrebbero contenere (grossi) errori di programmazione. Tuttavia nulla può essere dato per certo e, in ogni caso, non si può escludere che errori o eventi imprevisti, quali malfunzioni dell'hardware, virus, o altre cause materiali esterne (sbalzi di tensione, fulmini ecc...) compromettano l'integrità dei dati. Se accade che **Adsenze** non riesca a partire oppure, in esecuzione, produca il messaggio: "INTERRUZIONE IMMEDIATA DEL PROGRAMMA...", vuol dire che il *database* è rovinato; in questo caso non servirebbe a niente tentare di ricostruirlo con **Access**, anche conoscendolo perfettamente (i dati sono talmente in relazione fra loro che tentare di ricostruirli modificando separatamente ogni *record* sarebbe un'impresa disperata).

La soluzione del problema è, ancora una volta, di tipo "profilattico": è buona norma, anzi diremmo quasi che obbligatorio, salvare il *database* (meglio ancora se automaticamente) tutte le volte che si sta per partire con **Adsenze**. Nell'IRA abbiamo creato un piccolo file di comandi, di nome **Adsenze.bat**, che prima copia il *database* in un file con un altro nome e poi lancia **Adsenze.exe**, e abbiamo costruito sul *desktop* un'icona che è collegata ad **Adsenze.bat** anziché direttamente al programma. Anche se ciò produce l'apertura di una (fastidiosa) finestra Ms-Dos, che alla fine deve essere chiusa dall'operatore (e che spesso deve essere *iconizzata* per evitare che si inserisca indebitamente in primo piano), almeno in questo modo i dati sono protetti: se si verifica un errore, è sufficiente, dopo che se ne è scoperta e rimossa la causa, eliminare il *database*, dare il nome del *database* al file di salvataggio e ripartire reinserendo i dati dell'ultima esecuzione di **Adsenze**.

Consigliamo vivamente a tutti di fare altrettanto.

Ulteriori precisazioni sull'utilizzo dei programmi in rete

Nelle pagine 3, 5 e 45 del rapporto sulle "Procedure automatiche per la registrazione delle presenze e assenze del personale" si è fatto riferimento all'utilizzo dei programmi in una configurazione di rete di Personal Computer, e ai problemi che potrebbero insorgere per quello che riguarda la protezione dei dati. A seguito di verifiche effettuate dopo la stesura del citato rapporto (e ad alcune modifiche apportate al programma **Dipcheck**), riteniamo dover fornire ulteriori precisazioni:

- anzitutto la nuova versione di **Dipcheck** (modificato in modo da "aggirare" alcune impostazioni iniziali del *driver ODBC*), non crea files temporanei, e quindi può essere usata liberamente da chiunque e da qualunque punto della rete, purché il *database* sia visibile (in sola lettura);
- per quello che riguarda gli altri programmi, si conferma quanto detto, e precisamente:
 1. se il sistema in rete è gestito da **Windows NT Server** (che consente la protezione di singoli files), i programmi sono eseguibili da diversi punti della rete solo se la directory in cui risiedono è condivisa sia in lettura che in scrittura; in questo caso bisogna rendere i files dei programmi accessibili in sola lettura, e il file del *database* accessibile in sola lettura a tutti e in scrittura solo agli *Usernames* degli "addetti";
 2. se programmi e *database* risiedono in un PC con **Windows 95** (che può mettere in condivisione le directories ma non i singoli files), la directory in cui si trovano deve essere condivisa dalla rete in lettura (per far vedere il *database* a **Dipcheck**), ma non in scrittura, in modo che i programmi che modificano il *database* possano essere eseguiti soltanto nel PC in cui risiedono (che sarà di uso esclusivo degli "addetti").

Forse, se gli "addetti" sono una sola persona (o anche più persone, che non abbiano troppe difficoltà a lavorare su un solo PC), la seconda soluzione è più semplice e più sicura.