**PROGETTAZIONE ADT listaPrenotazione.c/.h**

La struttura è identica alla sua controparte per i corsi (liste.c) l’unica differenza sono la presenza di variabili Prenotazioni invece delle variabili Corso.

**PROGETTAZIONE FUNZIONI**

* newListPrenotazioni
* insertListPrenotazioni
* insertNode
* removeListPrenotati
* removeNode
* reverseListPrenotati
* cancellaPrenotazione
* getFirstPrenotazione
* isEmptyPrenotazione
* ricercaMesePrenotazione
* stampaListaPrenotazioni
* scriviFilePrenotazioni

Presentato la stessa progettazione della loro controparte per i corsi (liste.c), l’unica differenza sono la presenza di variabili Prenotazione invece delle variabili Corso.

ricercaListaPrenotati:

1. Controllo che la lista esista (caso negativo ritorno 0).
2. Creo una lista che conterrà i riscontri della ricerca (questo per gestire le prenotazioni di un cliente o di un corso).
3. Scorro ogni elemento della lista e tramite un selettore imposto il parametro di ricerca:

* Selettore=0 Ricerca per ID Cliente
* Selettore=1 Ricerca per ID Corso
* Selettore=2 Ricerca per ID Prenotazione

1. Quando trovo l’elemento lo inserisco nella nuova lista tramite il comando insertList, con posizione uguale a 0.
2. Al termine del ciclo ritorno la lista.

cancellaPrenotazioniDi:

1. Controllo che la lista esista (caso negativo ritorno 0).
2. Inizializzo una variabile contatore elementiCancellati=0, che mi servirà per tener conto di quante prenotazioni sono state cancellate a seguito della cancellazione di un cliente o della cancellazione del corso.
3. Scorro ogni elemento della lista è tramite il selettore passato come parametro imposto il parametro per scegliere cosa eliminare:

* Selettore=0 Cancello tutte le prenotazioni di un cliente
* Selettore=1 Cancello tutte le prenotazioni di un corso

1. Se trovo l’elemento, richiamo la funzione removeListPrenotati e incremento il contatore elementiCancellati.
2. Ritorno il contatore elementiCancellati.