

Un sistema per gestire un parcheggio a pagamento può essere realizzato in C++ con le strutture seguenti:

```
struct automob
{
    char istArr[LUN];
    int ident;
};
struct park
{
    automob vettAuto[DIM];
    int numAuto;
    int contaAuto;
};
```

Il parcheggio può contenere al massimo **DIM** automobili. Gli interi **numAuto** e **contaAuto** indicano, rispettivamente, il numero di auto effettivamente presenti nel parcheggio ed il numero totale di auto che sono state parcheggiate nel parcheggio dal momento dell'attivazione del sistema. Di ogni auto viene memorizzato l'istante di arrivo e un numero intero progressivo che identifica l'auto (la prima auto avrà numero 0, la seconda 1, e così via). L'istante di arrivo è espresso da una stringa nella forma **hh:mm**. Quando è creata un'istanza di tipo **park**, non è presente nessuna auto nel parcheggio.

Scrivere il corpo delle seguenti funzioni C++.

1. **bool arrivo(park* p, const char istArr[])** che implementa l'arrivo all'istante **istArr** di un'auto nel parcheggio indirizzato da **p**. Se il parcheggio non è al completo, l'auto viene inserita nella prima posizione libera del vettore **vettAuto** e la funzione restituisce **true**; altrimenti la funzione restituisce **false**.
2. **int intTem(const automob *pa, const char istTem[])** che restituisce la differenza in minuti tra l'istante di arrivo dell'auto individuata da **pa** e l'istante **istTem**. Nel realizzare la funzione si utilizzi la funzione **atoi()** che converte la stringa passata come parametro alla funzione in un numero intero (per esempio, la stringa "23" è convertita nel numero intero 23).
3. **bool partenza(park* p, const char istPar[], int ident, float* costo)** che implementa la partenza di un'auto dal parcheggio all'istante **istPar**. La funzione rimuove l'auto di identificatore **ident** dall'array **vettAuto**, ricompatta l'array mantenendolo ordinato, calcola il tempo in minuti che l'auto è stata nel parcheggio e restituisce il costo del parcheggio tramite il **float** puntato da **costo**. Nel determinare il costo si consideri che il parcheggio costa 0.50 Euro l'ora e la prima ora non è frazionabile. La funzione restituisce **false** se l'auto non si trova nel parcheggio; altrimenti restituisce **true**.
4. **void stampaOra(const park* p, const char ist[])** che stampa in ordine di arrivo gli identificatori delle auto che sono nel parcheggio da meno di un'ora rispetto all'istante **ist** passato come argomento alla funzione;
5. **void salva(const park* p, const char nome[])** che salva sul file il cui nome è passato come secondo argomento alla funzione, gli identificatori e gli istanti di arrivo di tutte le auto presenti nel parcheggio in ordine inverso rispetto al loro arrivo. Ogni auto occupa una riga nel file.

NOTE SULLO SVOLGIMENTO DELLA PROVA:

- PER SVOLGERE L'ELABORATO, APRIRE il Dev-C++ (dal Menù **Avvio** (o **Start**) nella barra degli strumenti in fondo allo schermo, selezionare Programmi e quindi Dev-C++);
- PRIMA DI INIZIARE LO SVOLGIMENTO DELL'ELABORATO, selezionare la voce **Identifica studente** nel menù **Strumenti** all'interno dell'ambiente Dev-C++ e inserire i dati richiesti;
- Per svolgere l'elaborato, aprire il progetto *esaInf.dev* presente nel directory *c:\esame\esaInf* e scrivere le funzioni richieste nel file *compito.cpp*, già presente nel progetto;
- per una corretta stampa dell'elaborato bisogna mantenere il codice entro i margini imposti dall'ambiente Dev-C++ (linea verticale presente alla destra della pagina);
- SE L'ELABORATO È STATO COMPILATO SENZA ERRORI, PRIMA DELLA CONSEGNA, selezionare la voce **Consegna Compito** nel menù **Strumenti** all'interno dell'ambiente Dev-C++ e premere il tasto INVIO fino a quando non sparisce la finestra che è stata attivata.