

## Prova scritta del 11/7/2011

Strutturare adeguatamente i programmi ed evidenziarne la strutturazione mediante indentazione. Inserire anche adeguati commenti

- 1) (a) Realizzare una funzione di nome `inverti` che, presi come suoi parametri un array di caratteri `A` e la sua dimensione `n`, inverte l'ordine dei caratteri in `A` (es., dato `A = ('r', 'o', 'm', 'a')` il nuovo valore di `A` sarà `('a', 'm', 'o', 'r')`).
- (b) Descrivere la funzione anche tramite un diagramma di flusso.

2) Scrivere un programma principale che richiede all'utente il nome di un file, legge dal file specificato una sequenza di caratteri, la memorizza in un array `V` (di capacità massima 1000 caratteri), ed utilizzando (obbligatoriamente) la funzione `inverti` realizzata al punto (1) determina la sequenza inversa a quella data e quindi provvede a memorizzarla in un nuovo file il cui nome è ottenuto dal nome del file di input a cui viene aggiunto il prefisso "inverso di" (ad esempio, se "dati.txt" è il file di input, il nome del nuovo file sarà "inverso di dati.txt"). Si assuma che i nomi dei file possano essere al massimo di 100 caratteri.

3) Sia `S` il tipo di una struttura dati costituita da un campo `Nome`, di tipo stringa (max. 64 caratteri), e un campo `Num`, di tipo intero, e sia `T` un'array di elementi di tipo `S` (max. 100). Scrivere un programma principale che permetta di eseguire le seguenti operazioni:

1. *inserisci*: richiede all'utente una stringa ed un numero e li memorizza, rispettivamente, nei campi `Nome` e `Num` del primo elemento libero di `T`; se non c'è spazio libero in `T`, stampa un messaggio d'errore (N.B. All'inizio gli elementi di `T` sono considerati tutti "liberi". La stringa letta può essere una stringa qualsiasi, anche contenente caratteri "spazio");
2. *ricerca*: richiede all'utente una stringa `N` e stampa il valore del campo `Num` dell'elemento di `T` il cui campo `Nome` vale `N` (se non lo trova stampa un messaggio d'errore);
3. *salva* `T` su un file di nome "dati\_salvati.txt";
4. *esci*.

Il programma presenta all'utente (su standard output) il menù delle possibili operazioni, esegue l'operazione scelta e quindi ripete dall'inizio finché non viene scelta l'operazione 4.

Esempio esecuzione (input sottolineati):

Scegli un'operazione (1. inserisci – 2. ricerca – 3. salva – 4. esci): 1  
 Dai un nome: carla  
 Dai un numero: 30  
 Scegli un'operazione (1. inserisci – 2. ricerca – 3. salva – 4. esci): 1  
 Dai un nome: mario  
 Dai un numero: 40  
 Scegli un'operazione (1. inserisci – 2. ricerca – 3. salva – 4. esci): 2  
 Dai un nome da cercare: mario  
 Numero per mario: 40  
 Scegli un'operazione (1. inserisci – 2. ricerca – 3. salva – 4. esci): 3  
 Dati salvati  
 Scegli un'operazione (1. inserisci – 2. ricerca – 3. salva – 4. esci): 4  
 Terminato.

N.B. Utilizzare soltanto stringhe "tipo C" (array di caratteri ...).

4) [SOLO CdL in FISICA] Dato il seguente programma C++

```
struct s {int a; s* b;};  
s* alfa(int x, s* y) {  
    s* t = new s;  
    t->a = x;  
    t->b = y;  
    return t;  
}  
int main() {  
    s* z;  
    z = alfa(5, NULL);  
    z = alfa(7, z);  
    return 0;  
}
```

illustrare mediante un disegno le strutture dati create durante la sua esecuzione. Giustificare dettagliatamente la risposta.