



Cognome: _____ Nome: _____ Matricola: _____ Voto: ____/30

| Quesito | 1 | 2 | 3 | 4.1 | 4.2 | 4.3 | Tot |
|---------------|---|---|---|-----|-----|-----|-----|
| Punteggio Max | 3 | 3 | 7 | 7 | 7 | 3 | 30 |
| Valutazione | | | | | | | |

Istruzioni:

- Il tempo massimo a disposizione per svolgere la prova è di 1ora e 45 minuti.
- È vietato consultare appunti e utilizzare calcolatrici, telefoni, PC o qualsiasi dispositivo elettronico.
- Il voto minimo per superare la prova è 18.

Esercizio 1. Codifica Binaria (3 punti)

1. Dati i due numeri $A = -21_{10}$ e $B = 32_{16}$, effettuare la conversione in base 2, notazione complemento a 2, sul numero minimo di bit necessari a rappresentare entrambi gli operandi. Si effettuino quindi le operazioni $A+B$ e $A-B$ indicando esplicitamente se si verifica overflow. **Si motivi la risposta e si mostrino i passaggi eseguiti.**

Esercizio 2. Funzioni Ricorsive (3 punti)

Si scriva una funzione ricorsiva che, ricevuto come parametro un array di interi e ogni parametro ritenuto necessario, individui e restituisca al chiamante il valore massimo dell'array.

Esercizio 3. Matrici (7 punti)

Sia data una matrice quadrata di dimensione massima N , $mat[N][N]$, che memorizza valori interi. Si definisca una funzione che, ricevuti in ingresso **almeno i seguenti parametri**

- la matrice *mat*
- il valore intero *dim_mat*, che rappresenta la dimensione effettiva della matrice *mat*
- un valore intero *dim_sm*
- un valore intero *val*

calcoli e restituisca al chiamante il numero di sottomatrici quadrate di dimensione *dim_sm* che hanno somma degli elementi esattamente uguale a *val*.

Esempio: dati i seguenti valori per la matrice *mat*, se il parametro *val* = 80

| | | | | |
|---|----|----|----|----|
| 1 | 3 | 10 | 18 | 17 |
| 2 | 1 | 9 | 8 | 8 |
| 3 | 5 | 7 | 7 | 7 |
| 4 | 10 | 5 | 9 | 20 |
| 6 | 11 | 11 | 1 | 0 |

l'unica sottomatrice di dimensione 3 con somma pari a *val* è quella evidenziata; pertanto la funzione restituirà il valore 1.

Si può assumere che il parametro *dim_sm* abbia sempre un valore minore di *dim_mat*, è cioè che la dimensione specificata per la sottomatrice sia strettamente minore di quella della matrice principale.

Esercizio 4 (17 punti)

Un file di testo contiene nomi e cognomi degli studenti di una classe, uno studente per riga, per esempio:

```
Maristella Matera  
Alessandra Matera  
Niccolò Izzo  
Maurizio Bianchi
```

Il numero di studenti memorizzati nel file non è noto a priori.

Punto 1 (7 punti). Si definisca una funzione, *ordine*, che riceve in ingresso il puntatore al file in cui sono memorizzati i nomi (e ogni altro parametro ritenuto necessario) e costruisce e restituisce al chiamante **una lista dinamica** che memorizza i nomi contenuti nel file:

- Ogni nodo memorizza il nome e il cognome di uno studente
- La lista dovrà essere costruita in modo da rispettare l'**ordinamento alfabetico dei cognomi**, quindi dei nomi nel caso di cognome uguale. Per esempio, per il file nell'esempio precedente, la lista memorizzerà i dati nel seguente ordine:

```
<Maurizio, Bianchi> → <Alessandra, Matera> → <Maristella, Matera> → <Niccolò, Izzo> - |
```

Prima di definire la funzione, **si definisca il tipo dei nodi e della lista.**

N.B.: È possibile usare funzioni predefinite per la gestione delle stringhe (funzioni della libreria *string* per il confronto tra e per l'assegnamento di stringhe).

Punto 2 (7 punti). Si definisca una seconda funzione, *statistiche*, che ricevuta in ingresso la lista costruita dalla funzione *ordine* (e ogni altro parametro ritenuto necessario) calcoli e restituisca il numero totale di studenti contenuti nella lista e il cognome che appare più volte nella lista e il numero delle sue occorrenze. Nel caso in cui ogni cognome sia presente una sola volta nella lista, la funzione restituisce il primo cognome nella lista e frequenza 1.

Punto 3 (3 punti). Si definisca il main del programma (completo di opportune dichiarazioni per il tipo del nodo della lista e della lista stessa e di ogni altra dichiarazione necessaria a livello globale e locale), che:

- riceve da **riga di comando** (parametro del main) il nome del file da cui leggere i nomi
- invoca la funzione *ordina*;
- invoca la funzione *statistiche* e stampa il numero totale di studenti, la parola con frequenza minima e il valore della frequenza, la parola con frequenza massima e il valore della frequenza.