

COGNOME _____ NOME _____ MATRICOLA _____

Durata parte teoria 45 minuti.

Esercizio A

Descrivere il QUICKSORT, riportando introduzione al problema, l'algoritmo di soluzione e la **relativa analisi di complessità debitamente motivata**.

Esercizio B

Si simuli l'esecuzione della funzione ricorsiva f, utilizzando i record di attivazione, e supponendo che la funzione venga richiamata con L1: 3→1→5→8→7→NULL e nella seguente situazione:

x=0; y=0; f(x,&y,L1);

```
f(int a, int * b, LINK lis) {  
    int x = 1;  
    if (lis != NULL) {  
        if (lis-->d < (a+(*b))  
        { *b = (*b)+1; printf("%d\n",(*b)); f(a+x,b, lis-->next); printf("%d\n",a); }  
        else if (lis-->d > (a+(*b))  
        {printf("%d\n",(*b)-x); *b=(*b)+2; f(a+x,b, lis-->next); printf("%d\n",a+x); }  
    }  
}
```

COGNOME _____ NOME _____ MATRICOLA _____

Durata: 2 ore

N. Esercizi: 4.

REGOLE GENERALI:

Per i seguenti esercizi predisporre:

- le strutture dati
- le variabili e le costanti
- Funzioni per il caricamento e salvataggio dei valori da file e/o tastiera (in base a anche a quanto richiesto dai singoli esercizi);
- Funzioni per la stampa a schermo dei risultati.
- Pulizia della memoria.
- **NON E' POSSIBILE CONTARE IL NUMERO DI NODI E/O UTILIZZARE QUALSIVOGLIA INFORMAZIONI SUL NUMERO DI NODI DI UNA LISTA.**

Esempio di struttura del main:

Main

1. ESERCIZIO 1 -> STAMPA A SCHERMO RISULTATO
2. ESERCIZIO 2 -> STAMPA A SCHERMO RISULTATO
3. CARICAMENTO DA FILE E/O TASTIERA -> ESERCIZIO 3 -> STAMPA A SCHERMO RISULTATO
4. CARICAMENTO DA FILE E/O TASTIERA -> ESERCIZIO 4 -> STAMPA A SCHERMO RISULTATO
5. Pulizia memoria

End

Mettere tutti i file .c (e eventuali .h o di testo contenenti le liste utilizzate) al termine nella cartella consegna nel formato cognome_MATRICOLA_nomefile.c

Compilare i sorgenti usando i flag -Werror -Wpedantic -Wextra

Esercizio 1 (4pt)

Data la seguente definizione di struttura:

```
struct citta {
    const char *nome;
    int popolazione;
};
```

Scrivere una funzione `secondLargest` che prenda in input un array di città e restituisca la seconda città più popolosa nell'array.

Esempio:

```
[{"Torino", 836000},
 {"Milano", 1371498},
 {"Roma", 4353738},
 {"Venezia", 249808}]    RISULTATO  {"Milano", 1371498}.
```

Esercizio 2 (4pt)

Il coefficiente binomiale $\binom{n}{k}$ che si legge ("n" su "k") è un numero intero non negativo definito dalla seguente formula:

$$C(n; k) = \binom{n}{k} = \frac{n!}{k! \cdot (n - k)!}, \quad n, k \in \mathbb{N}, 0 \leq k \leq n$$

COGNOME _____ NOME _____ MATRICOLA _____

Scrivere una funzione `coeffBinomiale` che calcoli il coefficiente binomiale usando la definizione ricorsiva $C(n;k) = C(n-1, k-1) + C(n-1, k)$; sapendo che $C(n, n) = C(n, 0) = 1$.

Esempio:

$n = 9, k = 3$ RISULTATO 84

$n = 10, k = 4$ RISULTATO 210

Esercizio 3 (7 punti)

Scrivere una funzione iterativa che date due liste in input e un intero k , calcoli la media delle somme dei nodi **successivi** al nodo di posizione k (qualora questi esistano).

Qualora una delle due liste non abbia più nodi **dopo la posizione k** , si assuma che i suoi nodi abbiano valore 0. Si stampi, per ogni iterazione $> k$, il valore della somma dei nodi corrispondenti.

Si analizzi e motivi la complessità in spazio e in tempo. (*Si consideri il primo nodo come nodo alla posizione 1*).

PREDISPORRE IL CARICAMENTO DA FILE DELLA LISTA al di fuori della funzione.

Esempio

L1: 1-->1-->2-->2-->2-->3-->3-->3-->3-->1-->1-->NULL

L2: 1-->1-->2-->2-->7-->7-->3-->3-->3-->NULL

$K = 5$

RISULTATO: 30

Esercizio 4 (7 punti)

Scrivere una funzione ricorsiva `invertiLista` che prenda in input una lista $L1$ e inverta l'ordine degli elementi della lista. Analizzare e motivare la complessità in spazio e tempo. **PREDISPORRE IL CARICAMENTO DA FILE DELLA LISTA al di fuori della funzione.**

Esempio L1: 1 --> 2 --> 3 --> 4 --> 5 --> NULL

RISULTATO 5 --> 4 --> 3 --> 2 --> 1 --> NULL.