

PROVA D'ESAME DI INFORMATICA II
14 DICEMBRE 2007

CORSO DI LAUREA IN MATEMATICA
UNIVERSITÀ DEGLI STUDI DI UDINE

Esercizio 1 - Punti 3. Valutare il comportamento asintotico della seguente funzione ricorsiva

$$T(n) = \begin{cases} 2T(n/2) + n^3 + n & \text{se } n > 1 \\ 1 & \text{se } n = 1 \end{cases}$$

Esercizio 2 - Punti 4. Calcolare con criterio di costo logaritmico il costo $T(n)$ in termini di tempo della seguente procedura

```
int Trip(int n){
    int i=n;
    int j=n;
    while(i>0){
        while(j>0){
            n=n+1;
            j=j-1;
        }
        j=n;
        i=i-1;
    }
    return n;
}
```

Quale è l'output della funzione Trip su un generico input?

Esercizio 3 - Punti 6. Dato un vettore A di interi si vuole determinare quale è l'elemento più ripetuto in A . Scrivere il codice di una funzione C per risolvere tale problema. Calcolarne la complessità con criterio di costo uniforme.

Esercizio 4 - Punti 6. Si considerino alberi in cui ogni nodo può avere un numero arbitrario di figli. Si implementino tali alberi associando ad ogni nodo la lista dei propri figli. Si descriva la struttura dati in C.

Scrivere il codice C di una funzione che visita tutti i nodi di un albero. Calcolarne la complessità con criterio di costo uniforme.

Esercizio 5 - Punti 6. Realizzare la struttura dati grafo con liste di adiacenza. Sia G un grafo orientato, connesso con n nodi. Sia Col un vettore di dimensione n che assegna ad ogni nodo colore verde oppure rosso. $Col[i]$ è l'intero 0 se il nodo i è verde, $Col[i]$ è l'intero 1 se il nodo i è rosso. Un cammino alternato in G è un cammino in cui si alternano nodi di colore rosso e nodi di colore verde. Dati due nodi u e v di G si vuole determinare la lunghezza del più corto cammino alternato

da u a v . Si scriva il codice di una funzione C che risolve questo problema. Se ne calcoli la complessità con criterio di costo uniforme.

Esercizio 6 - Punti 5. Descrivere una URM che calcola la funzione $f : \mathbb{N} \longrightarrow \mathbb{N}$ definita come

$$f(x) = 3 * x$$