

Algoritmi e Strutture Dati – 07/01/14

Esercizio 1 – Punti ≥ 5 (Parte A)

Calcolare la complessità della procedura `mystery()` descritto di seguito.

```

int mystery(int n)
{
    int i, j, s, k
    s = 0
    for i = 1 to n do
    {
        j = 1
        while j < n do
        {
            k = 1
            while k ≤ n do
            {
                s = s + 1
                k = k * 3
            }
            j = j * 2
        }
    }
    return s
}

```

Esercizio 2 – Punti ≥ 6 (Parte A)

Progettare un albero red-black di altezza nera almeno 1 in cui inserendo una chiave a vostra scelta cambia l'altezza nera dell'albero (ovvero, l'altezza nera della radice). Disegnare sia l'albero di partenza che l'albero risultante dall'operazione di inserimento.

Esercizio 3 – Punti ≥ 7 (Parte A)

La mappa di un videogioco è rappresentata da un grafo non orientato $G = (V, E)$: ogni nodo v rappresenta una "stanza", e ogni arco (u, v) indica l'esistenza di un passaggio diretto dalla stanza u alla stanza v . Le stanze sono in tutto n . L'array di booleani $M[1 \dots n]$ indica la presenza o meno di un mostro in ciascuna stanza. All'inizio del gioco, il giocatore si trova nella stanza s ; ad ogni turno può spostarsi dalla stanza corrente in una adiacente. Scopo del gioco è raggiungere una stanza destinazione d .

- Scrivere un algoritmo che restituisca la lunghezza del più breve cammino da s a d privo di mostri, se esiste; restituisca $+\infty$ se tale cammino non esiste
- Scrivere un algoritmo che restituisca il minor numero di mostri che è impossibile evitare andando da s a d .

Discutere informalmente la correttezza delle soluzioni proposte e calcolare la complessità computazionale.

Esercizio 4 – Punti ≥ 12 (Parte B)

Disponiamo di n libri di identico spessore, ma diverse altezze; il libro i -esimo ha altezza $y[i]$. Vogliamo disporre i libri su scaffali che possono contenere fino ad L libri. Gli scaffali vengono posizionati uno sopra l'altro in modo da non sprecare spazio in verticale. L'altezza di ciascuno scaffale è pari all'altezza massima dei libri che contiene. L'altezza complessiva occupata dall'intera collezione di libri è data dalla somma delle altezze dei singoli scaffali. Scrivere un algoritmo che, dati in input il vettore $y[1 \dots n]$ e il valore L , restituisce la minima altezza totale occupata dalla collezione di libri.

Discutere informalmente la correttezza della soluzione proposta e calcolare la complessità computazionale.