Cognome: **Nome**: **# Matricola**: **Riga**: **Col**:

Algoritmi e Strutture Dati - Parte 1 - 21/01/2019

Esercizio -1 Iscriversi allo scritto entro la scadenza. In caso di inadempienza, -1 al voto finale.

Esercizio 0 Scrivere correttamente nome, cognome, numero di matricola, riga e colonna su tutti i fogli consegnati. Consegnare foglio A4 e foglio protocollo di bella. In caso di inadempienza, -1 al voto finale.

Esercizio A1 – Punti ≥ 8

Trovare i limiti superiore e inferiore più stretti possibili per la seguente equazione di ricorrenza:

$$T(n) = \begin{cases} 4T(\lfloor n/4 \rfloor) + 9T(\lfloor n/9 \rfloor) + n\sqrt{n} & n > 9 \\ 1 & n \leq 9 \end{cases}$$

Esercizio A2 – Punti ≥ 10

Sia A un vettore ordinato contenente n interi positivi con valori potenzialmente ripetuti e sia v un valore intero.

Scrivere un algoritmo **int** ceil(**int**[ ] A, **int** n, **int** v) di complessità $O(\log n)$ che restituisca il più piccolo elemento del vettore che sia più grande di v, oppure -1 se tale valore non esiste.

Prestare attenzione ai casi particolari e agli indici.

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

Esempio: se $A = [1, 3, 3, 4, 5, 5, 8]$ e $v = 6$, il valore da restituire è 8; se $v = 8$, il valore da restituire è -1.

Esercizio A3 – Punti ≥ 12

Da bravi informatici, durante le scorse vacanze tutti noi abbiamo costruito un **albero binario** di Natale, a cui abbiamo appeso gli addobbi. Ogni nodo t dell'albero ha un addobbo caratterizzato da una certa "brillantezza", il cui valore può essere trovato nel campo $t.brilliance$.

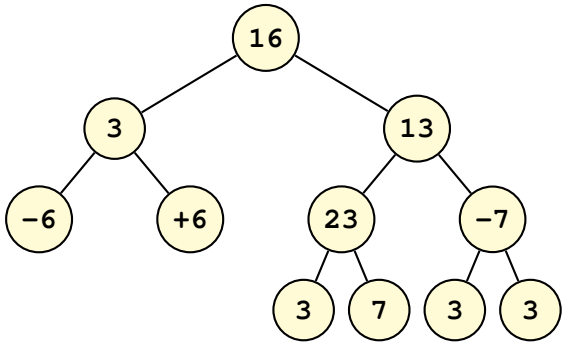
Ovviamente, non vogliamo un albero in cui tutti i nodi abbiano la stessa brillantezza. Nonostante l'opinione di mia moglie sui gusti degli informatici, questo sarebbe troppo noioso perfino per noi. Un certo ordine, però, ci vuole.

Un *albero binario di Natale* è un albero binario tale per cui vale la seguente regola: *tutti i livelli hanno la stessa brillantezza*. La brillantezza di un livello è pari alla somma della brillantezza dei nodi appartenenti a tale livello. Un albero binario **nil** è un albero binario di Natale, ma triste.

Scrivere un algoritmo che prende in input un albero binario T e restituisca **true** se T è un albero binario di Natale, **false** altrimenti.

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

Per esempio, l'albero seguente è un albero binario di Natale.



Cognome: **Nome**: **# Matricola**: **Riga**: **Col**:

Algoritmi e Strutture Dati - Parte 2 - 21/01/2019

Esercizio -1 Iscriversi allo scritto entro la scadenza. In caso di inadempienza, -1 al voto finale.

Esercizio 0 Scrivere correttamente nome, cognome, numero di matricola, riga e colonna su tutti i fogli consegnati. Consegnare foglio A4 e foglio protocollo di bella. In caso di inadempienza, -1 al voto finale.

Esercizio B1 – Punti ≥ 8

Alla prossima edizione di Google Hashcode, un brillante professore di Algoritmi del nord Italia vorrebbe distribuire ai partecipanti delle t-shirt[1].

Ha a disposizione un certo numero di t-shirt di 6 taglie diverse, numerate da 1 a 6, corrispondenti a XS,S,M,L,XL,XXL. Il vettore T contiene il numero di t-shirt per ogni taglia, ovvero $T[i]$ contiene il numero di magliette a disposizione per l'i-esima taglia.

Ogni partecipante può utilizzare magliette delle propria taglia o di una taglia superiore, le altre sono o troppo strette o troppo larghe. Il vettore P contiene le taglie dei partecipanti, ovvero se $P[j]$ contiene la taglia i, con $1 \leq i \leq 5$, il partecipante j può indossare la taglia i oppure $i+1$.

Il numero totale di partecipanti è pari a n, mentre il numero di totale di magliette è $m = \sum_{i=1}^{6} T[i]$, con $m \geq n$.

Scrivere un algoritmo che restituisca **true** se è possibile assegnare ad ogni partecipante j una maglietta di una taglia adatta $P[j]$ o $P[j]+1$, **false** altrimenti. Ovviamente, se $m > n$, ci saranno delle magliette in avanzo.

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

Esercizio B2 – Punti ≥ 10

Harmonyville è un villaggio **lineare** simile a Hateville (n case in fila numerate da 1 a n). Ad Harmonyville, tutti vanno talmente d'accordo che in vista della prossima sagra hanno deciso di ri-dipingere le case tutte dello stesso colore. Ma la sagra è a breve e bisogna lavorare in fretta.

Ogni casa i richiede un numero $H[i]$ di ore per essere ridipinta. Il lavoro viene suddiviso fra k pittori. Ogni pittore può lavorare solo ad un insieme **contiguo** di case.

Ad esempio, se $k = 3$ e $n = 6$, una possibile soluzione potrebbe essere dare le prime due case al pittore 1, le seconde due al pittore 2, le terze due al pittore 3. Se $H = \{2, 5, 1, 3, 2, 1\}$ è il tempo necessario per dipingere ognuna delle sei case, la suddivisione è la seguente: $\{2, 5\}, \{1, 3\}, \{2, 1\}$.

Assumendo che i pittori lavorino in parallelo, il *tempo di completamento* per una certa suddivisione è pari al numero di ore assegnate al pittore con più alto carico di lavoro. Nell'esempio precedente, il pittore 1 ha il carico più alto ($2 + 5 = 7$ ore assegnate) e quindi il tempo di completamento è pari a 7.

Esiste però la possibilità di dividere le case in modo più efficiente; ad esempio, la suddivisione $\{2\}, \{5, 1\}, \{3, 2, 1\}$ può essere completata in $5 + 1 = 3 + 2 + 1 = 6$ ore, un'ora in meno della suddivisione precedente. Una suddivisione è *minimale* se il suo tempo di completamento è il più basso possibile fra tutti i tempi di completamento. Questa suddivisione è minimale.

Scrivere un algoritmo che prenda in input il vettore H, il numero di case n e il numero di pittori k, e restituisca il tempo di completamento della suddivisione minimale. In questo esempio, 6.

Discutere complessità e correttezza dell'algoritmo proposto.

Esercizio B3 – Punti ≥ 12

Sia dato un vettore A di interi positivi distinti e un valore intero positivo v. Scrivere un algoritmo che stampi tutti i modi per ottenere v come somma di valori in A, anche utilizzando più volte lo stesso valore.

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

Ad esempio, se $A = [1, 3, 2, 4]$ e $v = 7$, i modi possibili sono i seguenti (l'ordine non è importante):

```
[1, 1, 1, 1, 1, 1, 1]
[1, 1, 1, 1, 1, 2]
[1, 1, 1, 1, 3]
[1, 1, 1, 2, 2]
[1, 1, 1, 4]
[1, 1, 3, 2]
[1, 3, 3]
[1, 2, 2, 2]
[1, 2, 4]
[3, 2, 2]
[3, 4]
```

[1] Viste le restrizioni su ogni tipo di acquisto, è un pazzo sognatore.