

Cognome.....

Algoritmi e Strutture Dati

Nome.....

Prova scritta del 14 giugno 2019

Matricola.....

TEMPO DISPONIBILE: 2 ore

Le risposte agli esercizi 1 e 2 devono essere scritte negli appositi riquadri su questo foglio (risposte scritte su altri fogli non saranno considerate). La soluzione dell'esercizio 3 va scritta su uno dei fogli di protocollo forniti. Le brutte copie NON devono essere consegnate. Ricordatevi di scrivere cognome e nome su TUTTI i fogli (inclusi quelli di brutta).

1. Considerate l'albero AVL che si ottiene inserendo uno dopo l'altro, nell'ordine indicato, i seguenti numeri a partire da un albero AVL inizialmente vuoto:
14 28 33 16 19 22 23 20

(a) Disegnate l'albero ottenuto	(b) Scrivete l'elenco dei valori dei nodi ottenuto mediante la visita in ampiezza
	(c) Scrivete l'elenco dei valori dei nodi ottenuto mediante la visita in profondità in ordine anticipato
	(d) Scrivete l'elenco dei valori dei nodi ottenuto mediante la visita in profondità in ordine simmetrico
	(e) Scrivete l'elenco dei valori dei nodi ottenuto mediante la visita in profondità in ordine posticipato

(f) L'albero ottenuto al punto (a) è un albero perfettamente bilanciato?

(g) Se al punto (f) avete risposto SI scrivete la definizione di albero perfettamente bilanciato. Se avete risposto NO indicate il contenuto di <u>tutti</u> i nodi che non rispettano la condizione di bilanciamento relativa agli alberi perfettamente bilanciati.	(h) Disegnate l'albero 2-3 che si ottiene inserendo il un albero 2-3 inizialmente vuoto la sequenza di numeri precedente, nell'ordine indicato.
--	---

2. La seguente sequenza di numeri, memorizzata in un array, deve essere ordinata in modo crescente:

14 33 28 16 25 22 23 20

(a) Supponete di ordinare la sequenza mediante l'algoritmo <code>radixSort</code> (rispetto alla base 10). Indicate il contenuto dell'array dopo la prima iterazione del ciclo principale dell'algoritmo.
(b) Supponete di ordinare la sequenza mediante l'algoritmo <code>selectionSort</code> . Indicate il contenuto dell'array dopo la terza iterazione del ciclo principale dell'algoritmo.
(c) Supponete di ordinare la sequenza mediante l'algoritmo <code>bubbleSort</code> . Indicate il contenuto dell'array dopo la terza iterazione del ciclo principale dell'algoritmo.
(d) Supponete di ordinare la sequenza mediante l'algoritmo <code>heapSort</code> . Indicate il contenuto dell'array immediatamente dopo avere effettuato lo scambio che colloca il numero 28 nella sua posizione finale all'interno dell'array.

3. Si vuole costruire un algoritmo che, dato un vettore V di interi, determini *tutti* i sottovettori di somma massima (indicata successivamente con *sommaMax*) formati solo da numeri pari o solo da numeri dispari.

A tale scopo, utilizzando la tecnica della programmazione dinamica, a partire dal vettore $V = (v_1, v_2, \dots, v_n)$, si calcola un vettore $C = (c_1, c_2, \dots, c_n)$, dove c_i è la somma del sottovettore di V di somma massima formato solo da numeri pari o solo da numeri dispari che termina in posizione i . Utilizzando il vettore C e il vettore V si ricavano facilmente *sommaMax* e i vettori richiesti.

Esempi

- Dato $V = (3, 5, -1, 8, -2, 4, 2, -1, 5, 7, 8, 2)$, si ottiene $C = (3, 8, 7, 8, 6, 10, 12, -1, 5, 12, 8, 10)$. I sottovettori richiesti sono $(8, -2, 4, 2)$, $(5, 7)$, con *sommaMax* = 12.
- Dato $V = (3, 3, -1, -7, 1, 5)$, si ottiene $C = (3, 6, 5, -2, 1, 6)$. I sottovettori richiesti sono $(3, 3)$, $(1, 5)$, con *sommaMax* = 6.
- Dato $V = (3, 3, -1, -7, 1, 5, 4, 2, -4, 2, 0, 2)$, si ottiene $C = (3, 6, 5, -2, 1, 6, 4, 6, 2, 4, 4, 6)$. I sottovettori richiesti sono $(3, 3)$, $(1, 5)$, $(4, 2)$, $(4, 2, -4, 2, 0, 2)$ con *sommaMax* = 6.

Cosa di richiede

Risolvete i seguenti punti *nell'ordine indicato*.

- Scrivete una o più formule che permettano di ricavare il generico valore c_i . Per $i > 1$, c_i deve essere calcolato in funzione di c_{i-1} , di v_i e, se utile, di altri valori di indice $< i$ nel vettore V o di altre informazioni relative alla parte di vettore V fino alla posizione i .
- Indicate come dal vettore C sia possibile ottenere il valore di *sommaMax*.
- Descrivete *sinteticamente a parole* e poi *ad alto livello* in pseudocodice, un algoritmo basato sulla tecnica di *programmazione dinamica* che ricevendo il vettore V , calcoli il vettore C e fornisca in uscita il valore di *sommaMax*.
- Fornite una stima in funzione di n del tempo totale utilizzato dalla parte di algoritmo scritta al punto (c).
- Descrivete *sinteticamente a parole* come ricavare dal vettore C e da *sommaMax* l'elenco di *tutti* i sottovettori di somma massima *sommaMax* formati solo da numeri pari o solo da numeri dispari.
- Fornite una stima del tempo totale utilizzato dalla parte di algoritmo descritta al punto (e) in funzione di n .

Note

- Le risposte devono essere adeguatamente giustificate.
- Non cambiate i nomi stabiliti nel testo dell'esercizio, in particolare V , C e *sommaMax*.

Cognome.....

Algoritmi e Strutture Dati

Nome.....

Prova scritta del 14 giugno 2019

Matricola.....

TEMPO DISPONIBILE: 2 ore

Le risposte agli esercizi 1 e 2 devono essere scritte negli appositi riquadri su questo foglio (risposte scritte su altri fogli non saranno considerate). La soluzione dell'esercizio 3 va scritta su uno dei fogli di protocollo forniti. Le brutte copie NON devono essere consegnate. Ricordatevi di scrivere cognome e nome su TUTTI i fogli (inclusi quelli di brutta).

1. Considerate l'albero AVL che si ottiene inserendo uno dopo l'altro, nell'ordine indicato, i seguenti numeri a partire da un albero AVL inizialmente vuoto:
16 30 35 18 21 24 25 22

(a) Disegnate l'albero ottenuto	(b) Scrivete l'elenco dei valori dei nodi ottenuto mediante la visita in ampiezza
	(c) Scrivete l'elenco dei valori dei nodi ottenuto mediante la visita in profondità in ordine anticipato
	(d) Scrivete l'elenco dei valori dei nodi ottenuto mediante la visita in profondità in ordine simmetrico
	(e) Scrivete l'elenco dei valori dei nodi ottenuto mediante la visita in profondità in ordine posticipato

- (f) L'albero ottenuto al punto (a) è un albero perfettamente bilanciato?

(g) Se al punto (f) avete risposto SI scrivete la definizione di albero perfettamente bilanciato. Se avete risposto NO indicate il contenuto di <u>tutti</u> i nodi che non rispettano la condizione di bilanciamento relativa agli alberi perfettamente bilanciati.	(h) Disegnate l'albero 2-3 che si ottiene inserendo il un albero 2-3 inizialmente vuoto la sequenza di numeri precedente, nell'ordine indicato.
--	---

2. La seguente sequenza di numeri, memorizzata in un array, deve essere ordinata in modo crescente:

16 35 30 18 27 24 25 22

(a) Supponete di ordinare la sequenza mediante l'algoritmo <code>radixSort</code> (rispetto alla base 10). Indicate il contenuto dell'array dopo la prima iterazione del ciclo principale dell'algoritmo.
(b) Supponete di ordinare la sequenza mediante l'algoritmo <code>selectionSort</code> . Indicate il contenuto dell'array dopo la terza iterazione del ciclo principale dell'algoritmo.
(c) Supponete di ordinare la sequenza mediante l'algoritmo <code>bubbleSort</code> . Indicate il contenuto dell'array dopo la terza iterazione del ciclo principale dell'algoritmo.
(d) Supponete di ordinare la sequenza mediante l'algoritmo <code>heapSort</code> . Indicate il contenuto dell'array immediatamente dopo avere effettuato lo scambio che colloca il numero 30 nella sua posizione finale all'interno dell'array.

3. Si vuole costruire un algoritmo che, dato un vettore V di interi, determini *tutti* i sottovettori di somma massima (indicata successivamente con *sommaMax*) formati solo da numeri pari o solo da numeri dispari.

A tale scopo, utilizzando la tecnica della programmazione dinamica, a partire dal vettore $V = (v_1, v_2, \dots, v_n)$, si calcola un vettore $C = (c_1, c_2, \dots, c_n)$, dove c_i è la somma del sottovettore di V di somma massima formato solo da numeri pari o solo da numeri dispari che termina in posizione i . Utilizzando il vettore C e il vettore V si ricavano facilmente *sommaMax* e i vettori richiesti.

Esempi

- Dato $V = (3, 5, -1, 8, -2, 4, 2, -1, 5, 7, 8, 2)$, si ottiene $C = (3, 8, 7, 8, 6, 10, 12, -1, 5, 12, 8, 10)$. I sottovettori richiesti sono $(8, -2, 4, 2)$, $(5, 7)$, con *sommaMax* = 12.
- Dato $V = (3, 3, -1, -7, 1, 5)$, si ottiene $C = (3, 6, 5, -2, 1, 6)$. I sottovettori richiesti sono $(3, 3)$, $(1, 5)$, con *sommaMax* = 6.
- Dato $V = (3, 3, -1, -7, 1, 5, 4, 2, -4, 2, 0, 2)$, si ottiene $C = (3, 6, 5, -2, 1, 6, 4, 6, 2, 4, 4, 6)$. I sottovettori richiesti sono $(3, 3)$, $(1, 5)$, $(4, 2)$, $(4, 2, -4, 2, 0, 2)$ con *sommaMax* = 6.

Cosa di richiede

Risolvete i seguenti punti *nell'ordine indicato*.

- Scrivete una o più formule che permettano di ricavare il generico valore c_i . Per $i > 1$, c_i deve essere calcolato in funzione di c_{i-1} , di v_i e, se utile, di altri valori di indice $< i$ nel vettore V o di altre informazioni relative alla parte di vettore V fino alla posizione i .
- Indicate come dal vettore C sia possibile ottenere il valore di *sommaMax*.
- Descrivete *sinteticamente a parole* e poi *ad alto livello* in pseudocodice, un algoritmo basato sulla tecnica di *programmazione dinamica* che ricevendo il vettore V , calcoli il vettore C e fornisca in uscita il valore di *sommaMax*.
- Fornite una stima in funzione di n del tempo totale utilizzato dalla parte di algoritmo scritta al punto (c).
- Descrivete *sinteticamente a parole* come ricavare dal vettore C e da *sommaMax* l'elenco di *tutti* i sottovettori di somma massima *sommaMax* formati solo da numeri pari o solo da numeri dispari.
- Fornite una stima del tempo totale utilizzato dalla parte di algoritmo descritta al punto (e) in funzione di n .

Note

- Le risposte devono essere adeguatamente giustificate.
- Non cambiate i nomi stabiliti nel testo dell'esercizio, in particolare V , C e *sommaMax*.