PRENOTAZIONI
STUDIUM
PROVA ITINERE
DEL 13/12/21

→

SCADENZA
MERCOLEDI 8/12/21 ore 12:00

“ALGORITMI E COMPLESSITÀ”

CORSO DI LAUREA MAGISTRALE IN INFORMATICA

UNIVERSITÀ DEGLI STUDI DI CATANIA

ANNO ACCADEMICO 2017/18

Prima prova in itinere – 13 dicembre 2017

Si svolgano i seguenti esercizi, argomentando adeguatamente le risposte.

ESERCIZIO 1 (Analisi ammortizzata)

Utilizzando i metodi dell'aggregazione e del potenziale, si determini il costo ammortizzato per operazione di una sequenza di n operazioni, ove il costo c_i dell'i-esima operazione sia dato da

$$c_i = \begin{cases} 12 \cdot i & \text{se } i \text{ è potenza esatta di 5} \\ 7 & \text{altrimenti}\,. \end{cases}$$

ESERCIZIO 2 (Splay trees)

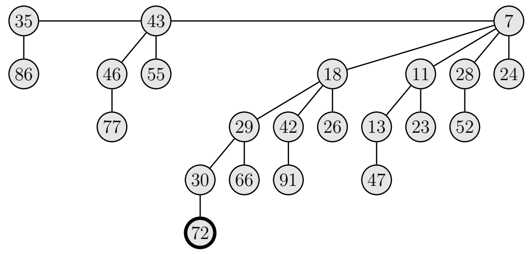
Si descrivano le operazioni di *zig-zag*, *zig-zig* e *zig* in uno splay tree di tipo bottom-up.
Quindi si eseguano nell'ordine dato le seguenti operazioni su uno splay tree la cui configurazione iniziale è quella di un albero binario completo contenente le 10 chiavi $\{4i : 1 \leq i \leq 10\}$:

- Search 20, 40
- Delete 24
- Insert 30

Nota bene: Si ricorda che un albero binario si dice *completo* quando tutti i suoi livelli, con al più l'eccezione dell'ultimo, sono completi e tutti i nodi nell'ultimo livello si trovano il più a sinistra possibile.

ESERCIZIO 3 (Heap binomiali)

(a) Si definiscano gli *heap binomiali* e si descrivano le operazioni DecreaseKey, ExtractMin, Delete e Insert. Quindi si cancelli il nodo evidenziato (contenente la chiave 72) dall'heap binomiale a lato e poi si inseriscano in successione le chiavi 10, 30 e 5.



(b) Si determinino un limite superiore ed un limite inferiore per il numero di alberi binomiali in un heap binomiale con n chiavi.

(c) Nel caso degli heap binomiali, è richiesto che gli alberi binomiali nella lista delle radici siano *ordinati* per grado. Perché?

ESERCIZIO 4 (B-trees)

Si definisca la struttura dati dei B-tree.

ESERCIZIO 1 **(Analisi ammortizzata)**

Utilizzando i metodi dell'aggregazione e del potenziale, si determini il costo ammortizzato per operazione di una sequenza di n operazioni, ove il costo c_i dell'i-esima operazione sia dato da

$$c_i = \begin{cases} 12\cdot i & \text{se } i \text{ è potenza esatta di 5} \\ 7 & \text{altrimenti}. \end{cases}$$

AGGREGAZIONE

$$T(n) = \sum_{i=1}^{n} c_i = \sum_{\substack{i=1 \\ i\in 5^{\mathbb{N}}}}^{n} (12\cdot i) + \sum_{\substack{i=1 \\ i\notin 5^{\mathbb{N}}}}^{n} 7$$

$i \in 5^{\mathbb{N}}$ $\quad 5^j$

$i = 1 \dots n$ $\quad j = 0, \dots, \lfloor \log_5 n \rfloor$

$$= 12 \sum_{j=0}^{\lfloor \log_5 n\rfloor} 5^j + 7(n - \lfloor \log_5 n \rfloor - 1)$$

$$= 12 \cdot \frac{5^{\lfloor \log_5 n\rfloor + 1} - 1}{5-1} + 7(n - \lfloor \log_5 n \rfloor - 1)$$

$$= 3\cdot\left(5^{\lfloor \log_5 n\rfloor + 1} - 1\right) + 7(n - \lfloor \log_5 n \rfloor - 1)$$

$$< 3 \cdot 5^{\log_5 n + 1} + 7n$$

$$= 3\cdot 5\cdot n + 7n$$

$$= 22\,n$$

$$\hat{c}_i = \frac{T(n)}{n} < 22$$

POTENZIALE

$$i = 5^k \longrightarrow 12\cdot 5^k$$

$$\frac{12\cdot 5^k}{5^k - 5^{k-1}} = \frac{\cancel{12}^{\,3}\cdot 5^{\cancel{k}}}{\cancel{5^{k-1}}\,\cancel{(5-1)}} = 15$$

30

$$\phi_i = \begin{cases} 0 & \text{se } i = 0 \\ 15\left(i - 5^{\lfloor \log_5 i \rfloor}\right) & \text{se } i \geq 1 \end{cases}$$

$$25 = 5^{\lfloor \log_5 30 \rfloor} = 5^2$$

$\phi_i \geq \phi_0 = 0$ — ovvio per $i = 0$

— per $i \geq 1 \rightarrow 5^{\lfloor \log_5 i \rfloor} \leq i \longrightarrow \phi_i = 15\left(i - 5^{\lfloor \log_5 i \rfloor}\right) \geq 0$

$\longrightarrow \sum_{i=1}^{m} c_i \leq \sum_{i=1}^{m} \hat{c}_i$.

$i = 1$

$\hat{c}_1 = c_1 + \phi_1 - \phi_0$

$= 12 + 0 - 0$

$= 12$

$125 = 5^3$

$\lfloor \log_5 124 \rfloor = 2$

$i \in 5^{\mathbb{N}}$ & $i \neq 1$

$i \in 5^{\mathbb{N}}$ $\quad \lfloor \log_5 i \rfloor = \log_5 i$

$5^{\lfloor \log_5 i \rfloor} = 5^{\log_5 i} = i$

$\lfloor \log_5 (i-1) \rfloor = \log_5 i - 1$

$5^{\lfloor \log_5 (i-1) \rfloor} = 5^{\log_5 i - 1} = \frac{i}{5}$

$\hat{c}_i = c_i + \phi_i - \phi_{i-1}$

$= 12 i + 15\left(\cancel{i} - \cancel{5^{\lfloor \log_5 i \rfloor}}\right) - 15\left(i - 1 - 5^{\lfloor \log_5 (i-1) \rfloor}\right)$

$= 12 i - 15\left(i - 1 - \frac{i}{5}\right)$

$= \cancel{12 i} - \overset{3}{\cancel{15}} \cancel{\frac{4}{5} i} + 15$

$= 15$

$i \notin 5^{\mathbb{N}}$

$\hat{c}_i = c_i + \phi_i - \phi_{i-1}$

$= 7 + 15\left(\cancel{i} - \cancel{5^{\lfloor \log_5 i \rfloor}}\right) - 15\left(\cancel{i} - 1 - \cancel{5^{\lfloor \log_5 (i-1) \rfloor}}\right)$

$= 7 + 15$

$i \notin 5^{\mathbb{N}} \longrightarrow 5^{\lfloor \log_5 i \rfloor} = 5^{\lfloor \log_5 (i-1) \rfloor}$

$= 22$

$$\hat{c}_i = \begin{cases} 12 & \text{se } i = 1 \\ 15 & \text{se } i \in 5^{\mathbb{N}} \ \& \ i \neq 1 \\ 22 & \text{se } i \notin 5^{\mathbb{N}} \end{cases}$$

ACCANTONAMENTI

i	c_i	$\hat{c}_i$
1	12	12
2	7	$7 + 15^{5}$
3	7	$7 + 15^{5}$
4	7	$7 + 15^{5}$
5	$12 \cdot 5$	15
6	7	$7 + 15^{25}$
7	7	$7 + 15^{25}$
$\vdots$	$\vdots$	$\vdots$
23	7	$7 + 15^{25}$
24	7	$7 + 15^{25}$
25	$12 \cdot 25$	15
26	7	$7 + 15^{125}$
27	7	$7 + 15^{125}$
$\vdots$	$\vdots$	$\vdots$

$$c_i = \begin{cases} 12 \cdot i & \text{se } i \text{ è potenza esatta di 5} \\ 7 & \text{altrimenti}\,. \end{cases}$$

$$15 \cdot (24 - 6 + 1) + 15 = 15 \cdot 20$$

$$= 3 \cdot 5 \cdot 4 \cdot 5$$

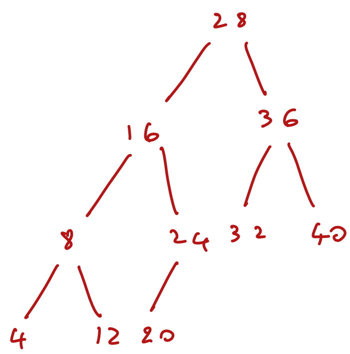
$$= 12 \cdot 25$$

ESERCIZIO 2 (Splay trees)

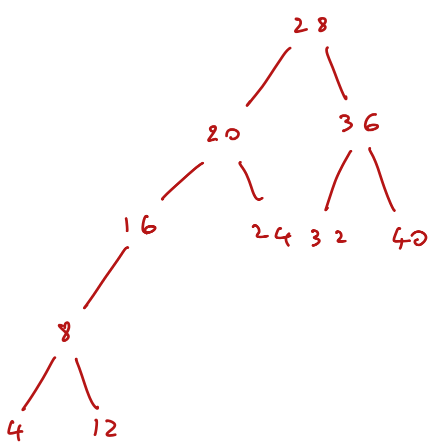
Si descrivano le operazioni di *zig-zag*, *zig-zig* e *zig* in uno splay tree di tipo bottom-up.
Quindi si eseguano nell'ordine dato le seguenti operazioni su uno splay tree la cui configurazione iniziale è quella di un albero binario completo contenente le 10 chiavi $\{4i : 1 \leq i \leq 10\}$:

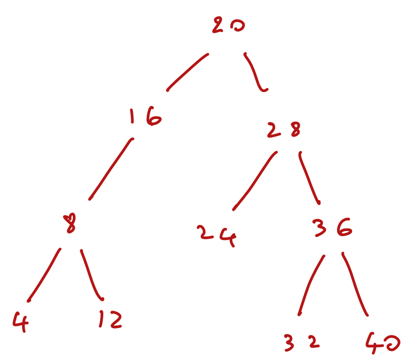
- Search 20, 40
- Delete 24
- Insert 30

Nota bene: Si ricorda che un albero binario si dice *completo* quando tutti i suoi livelli, con al più l'eccezione dell'ultimo, sono completi e tutti i nodi nell'ultimo livello si trovano il più a sinistra possibile.

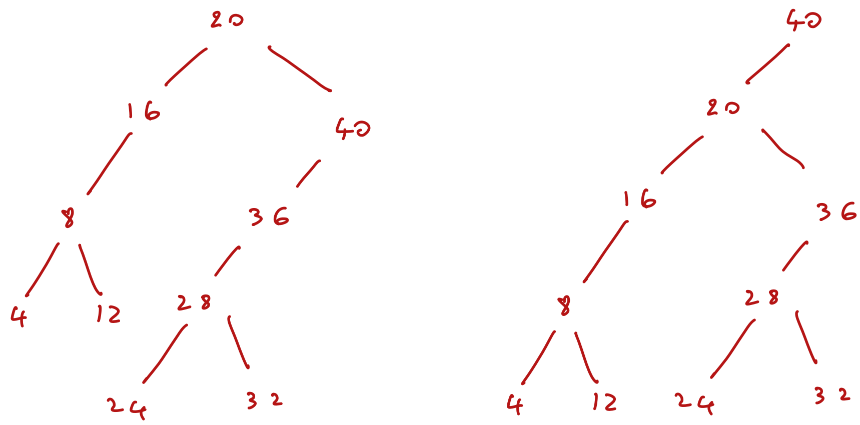


SEARCH 20

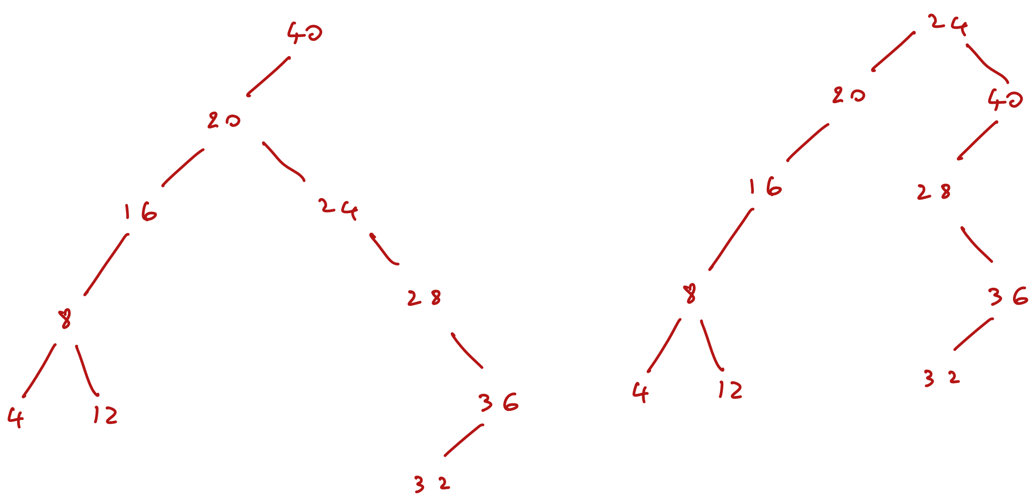


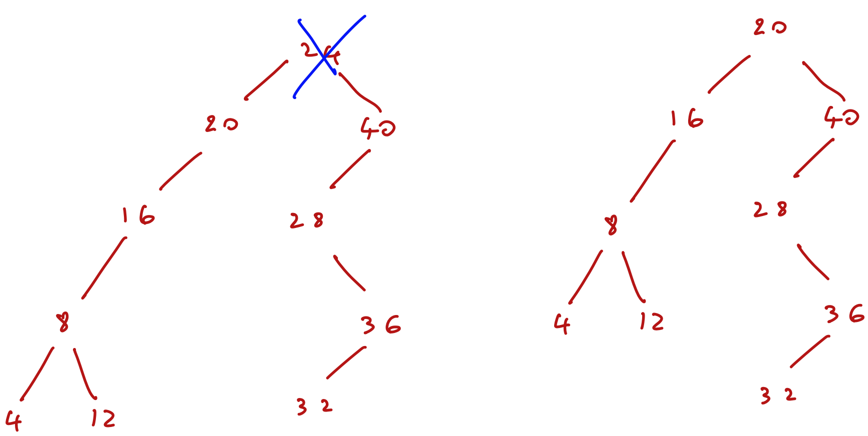


SEARCH 40

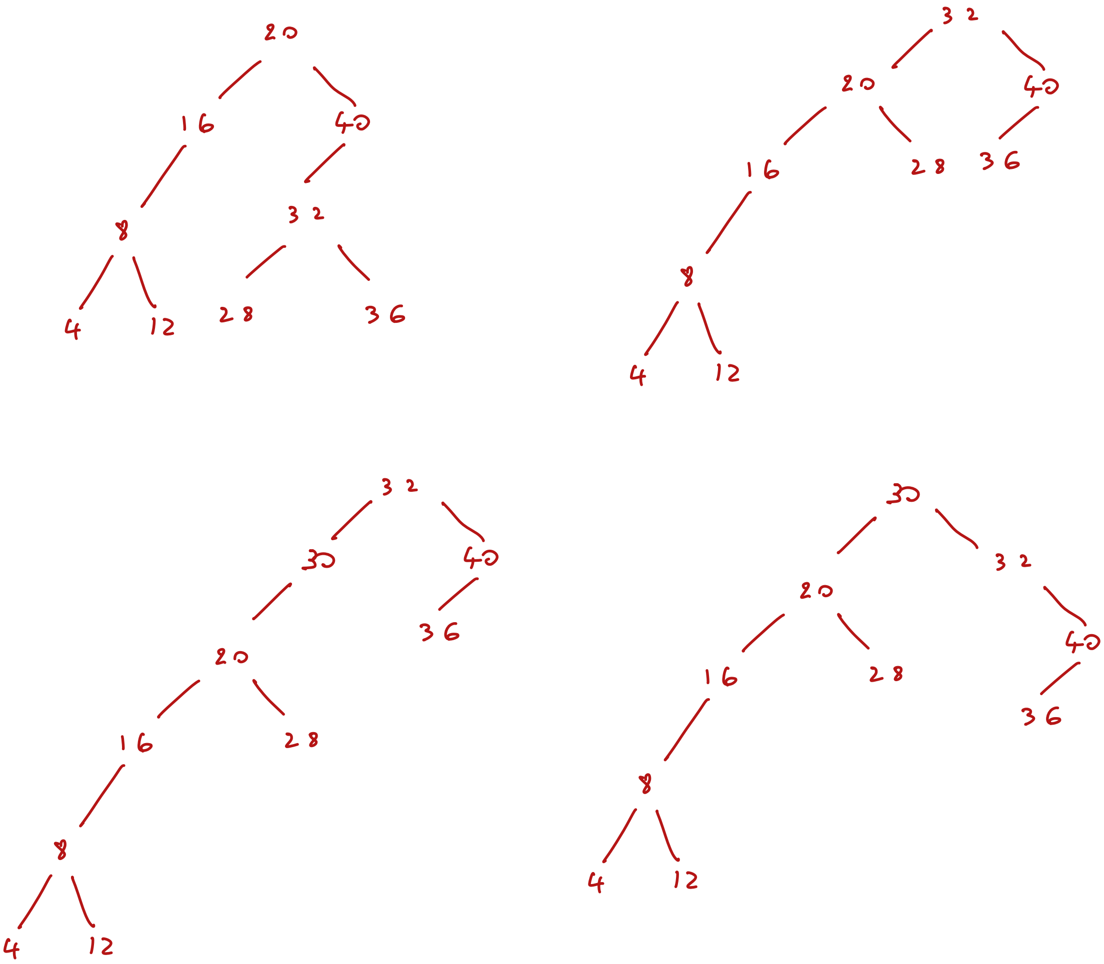


DELETE 24



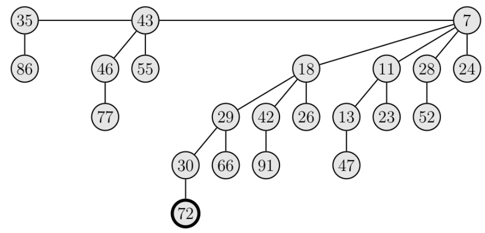


INSERT 30


20
16
40
8
32
4
12
28
36
32
20
40
16
28
36
8
4
12
32
30
40
20
36
16
28
8
4
12
30
20
32
16
28
40
36
8
4
12

ESERCIZIO 3 **(Heap binomiali)**

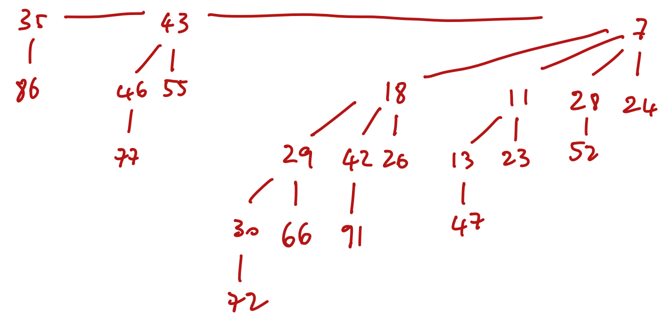
(a) Si definiscano gli *heap binomiali* e si descrivano le operazioni DecreaseKey, ExtractMin, Delete e Insert. Quindi si cancelli il nodo evidenziato (contenente la chiave 72) dall'heap binomiale a lato e poi si inseriscano in successione le chiavi 10, 30 e 5.

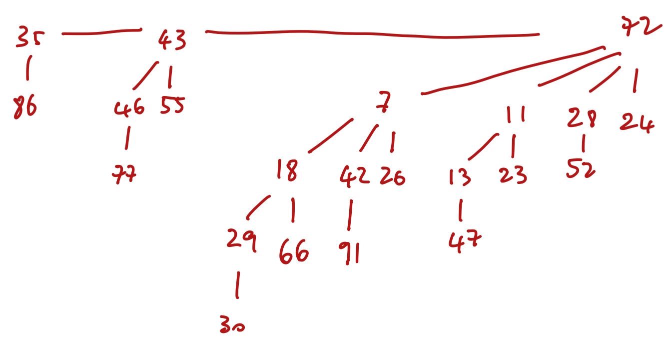


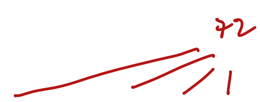
(b) Si determinino un limite superiore ed un limite inferiore per il numero di alberi binomiali in un heap binomiale con n chiavi.

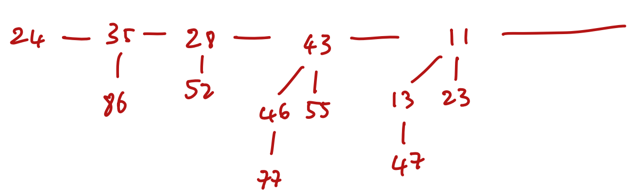
(c) Nel caso degli heap binomiali, è richiesto che gli alberi binomiali nella lista delle radici siano *ordinati* per grado. Perché?

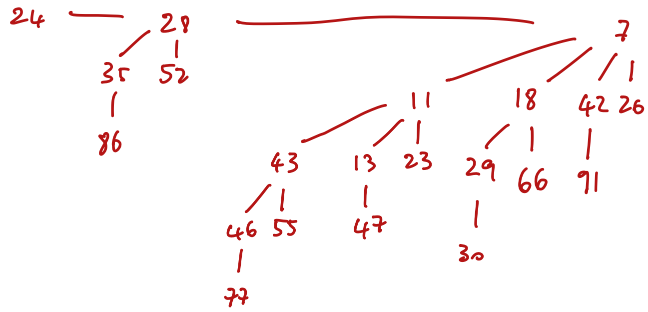
DELETE 72



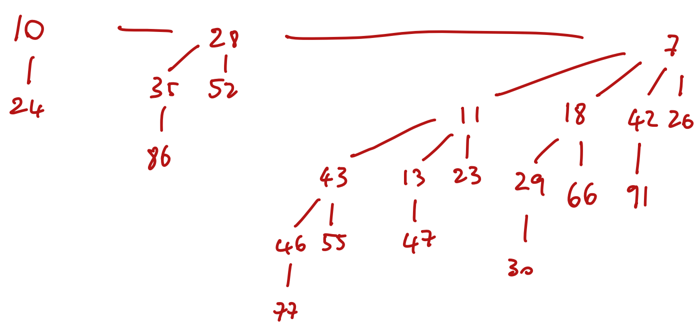




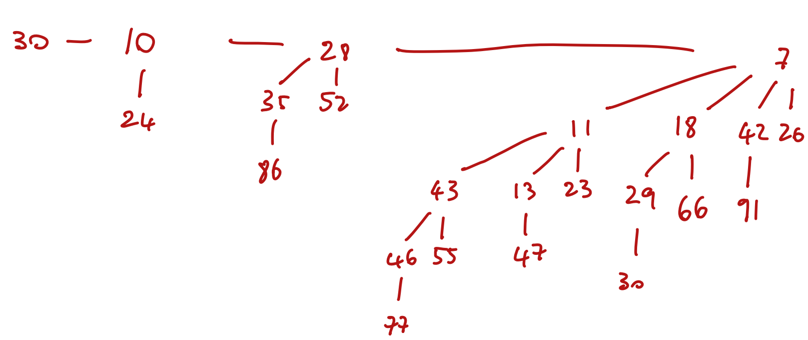




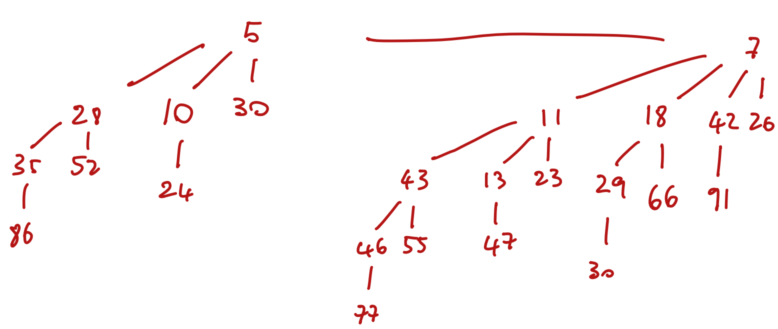
INSERT 10



INSERT 30



INSERT 5



$n = 0$ # alberi bin. $= 0$

$n > 0$ grado max $\lfloor \lg n \rfloor$

$1 \leq$ # alberi binomiali $\leq \lfloor \lg n \rfloor + 1$

"ALGORITMI E COMPLESSITÀ"
CORSO DI LAUREA MAGISTRALE IN INFORMATICA
UNIVERSITÀ DEGLI STUDI DI CATANIA
ANNO ACCADEMICO 2016/17

1^a prova in itinere – 14 dicembre 2016

Si svolgano i seguenti esercizi, argomentando adeguatamente le risposte.

ESERCIZIO 1

Utilizzando i metodi dell'**aggregazione** e del **potenziale**, si determini il costo ammortizzato per operazione di una sequenza di n operazioni, ove il costo dell'i-esima operazione sia dato da

(a) $c_i = \begin{cases} 8 \cdot i & \text{se } i \text{ è potenza esatta di } 5 \\ 7 & \text{altrimenti} \end{cases}$

oppure da

(b) *(Facoltativo)* $c'_i = \begin{cases} 8 \cdot i + 3 & \text{se } i \text{ è potenza esatta di } 5 \\ 10 & \text{altrimenti}\,. \end{cases}$

ESERCIZIO 2

(a) Si definisca la struttura dati dei *B-tree*.

(b) Si forniscano un limite inferiore e un limite superiore (quest'ultimo con dimostrazione) al numero di chiavi in un B-tree di grado minimo t e altezza h.

Da questi si deducano

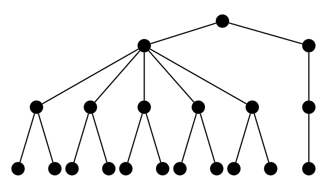
(b.1) un limite inferiore e un limite superiore all'altezza di un B-tree di grado minimo t contenente n chiavi;

(b.2) un limite inferiore e un limite superiore al grado minimo di un B-tree di altezza h contenente n chiavi.

ESERCIZIO 3

(a) Si enunci e si dimostri un lemma che fornisca una minorazione dei gradi dei figli di ciascun nodo in un heap di Fibonacci.

(b) Si stabilisca se possa esistere un heap di Fibonacci avente la struttura dell'albero a lato.



ESERCIZIO 4

Si descrivano le operazioni di *zig-zag*, *zig-zig* e *zig* in uno splay tree di tipo bottom-up.
Quindi si eseguano nell'ordine dato le seguenti operazioni su uno splay tree la cui configurazione iniziale è quella di un albero binario completo contenente le 9 chiavi $\{3i : 1 \leq i \leq 9\}$:

- Search 3, 9, 21
- Insert 16
- Delete 12
- Search 27

Nota bene: Si ricorda che un albero binario si dice *completo* quando tutti i suoi livelli, con al più l'eccezione dell'ultimo, sono completi e tutti i nodi nell'ultimo livello si trovano il più a sinistra possibile.

ESERCIZIO 1

Utilizzando i metodi dell'**aggregazione** e del **potenziale**, si determini il costo ammortizzato per operazione di una sequenza di n operazioni, ove il costo dell'i-esima operazione sia dato da

(a) $c_i = \begin{cases} 8 \cdot i & \text{se } i \text{ è potenza esatta di } 5 \\ 7 & \text{altrimenti} \end{cases}$

oppure da

(b) *(Facoltativo)* $c'_i = \begin{cases} 8 \cdot i + 3 & \text{se } i \text{ è potenza esatta di } 5 \\ 10 & \text{altrimenti}\,. \end{cases}$

AGGREGAZIONE

$$T(n) = \sum_{i=1}^{n} c_i = \sum_{\substack{i=1 \\ i \in 5^{\mathbb{N}}}}^{n} (8 \cdot i) + \sum_{\substack{i=1 \\ i \notin 5^{\mathbb{N}}}}^{n} 7$$

$$\leq 8 \sum_{j=0}^{\lfloor \log_5 n \rfloor} 5^j + 7n$$

$$= 8 \cdot \frac{5^{\lfloor \log_5 n \rfloor + 1} - 1}{5 - 1} + 7n$$

$$< 8 \cdot \frac{5n}{4} + 7n$$

$$= 10n + 7n$$

$$= 17n$$

$$\hat{c}_i = \frac{T(n)}{n} < \frac{17n}{n} = 17\,.$$

POTENZIALE

$$i = 5^k \qquad \frac{8 \cdot 5^k}{5^k - 5^{k-1}} = \frac{8 \cdot 5^k}{5^{k-1}(5-1)} = 10$$

$$\Phi_i = \begin{cases} 0 & i = 0 \\ 10\left(i - 5^{\lfloor \log_5 i \rfloor}\right) & i \geq 1 \end{cases}$$

$\Phi_i \geq 0 \quad \forall i$, in quanto $\quad i \geq 5^{\lfloor \log_5 i \rfloor}$

$\longrightarrow \quad \sum c_i \leq \sum \hat{c}_i$

$\underline{i=1}$

$\hat{c}_1 = c_1 + \phi_1 - \phi_0$

$= 8$

$\underline{i \in 5^N \;\&\; i \neq 1}$

$\hat{c}_i = c_i + \phi_i - \phi_{i-1}$

$= 8i + 10\left(\cancel{i} - \cancel{5^{\lfloor \log_5 i \rfloor}}\right) - 10\left(i-1 - 5^{\lfloor \log_5 (i-1) \rfloor}\right)$

$= \cancel{8i} - 10\cancel{i} + 10 + \cancel{\overset{2}{10} \cdot \frac{i}{5}}$

$= 10$

$i \notin 5^N$

$\hat{c}_i = c_i + \phi_i - \phi_{i-1}$

$= 7 + 10\left(\cancel{i} - \cancel{5^{\lfloor \log_5 i \rfloor}}\right) - 10\left(\cancel{i}-1 - \cancel{5^{\lfloor \log_5 (i-1) \rfloor}}\right)$

$= 17$

$$\hat{c}_i = \begin{cases} 8 & \text{se } i = 1 \\ 10 & \text{se } i \in 5^N \;\&\; i \neq 1 \\ 17 & \text{se } i \notin 5^N \end{cases}$$

ACCANTONAMENTI

i	c_i	$\hat{c}_i$
1	8	8
2	7	$7 + 10^{5}$
3	7	$7 + 10^{5}$
4	7	$7 + 10^{5}$
5	$8 \cdot 5$	10
6	7	$7 + 10^{25}$
7	7	$7 + 10^{25}$
$\vdots$	$\vdots$	$\vdots$
23	7	$7 + 10^{25}$
24	7	$7 + 10^{25}$
25	$8 \cdot 25$	10
26	7	$7 + 10^{125}$
27	7	$7 + 10^{125}$
$\vdots$	$\vdots$	$\vdots$

$$c_i = \begin{cases} 8 \cdot i & \text{se } i \text{ è potenza esatta di } 5 \\ 7 & \text{altrimenti} \end{cases}$$

$$\hat{c}_i = \begin{cases} 8 & \text{se } i = 1 \\ 10 & \text{se } i \in 5^{\mathbb{N}} \ \& \ i \neq 1 \\ 17 & \text{se } i \notin 5^{\mathbb{N}} \end{cases}$$

ESERCIZIO 2

(a) Si definisca la struttura dati dei *B-tree*.

(b) Si forniscano un limite inferiore e un limite superiore (quest'ultimo con dimostrazione) al numero di chiavi in un B-tree di grado minimo t e altezza h.

Da questi si deducano

(b.1) un limite inferiore e un limite superiore all'altezza di un B-tree di grado minimo t contenente n chiavi;

(b.2) un limite inferiore e un limite superiore al grado minimo di un B-tree di altezza h contenente n chiavi.

profondità	# nodi	# chiavi
0	1	1
1	2	$2(t-1)$
2	$2t$	$2t(t-1)$
3	$2t^2$	$2t^2(t-1)$
⋮	⋮	⋮
h	$2t^{h-1}$	$2t^{h-1}(t-1)$

$$1 + 2(t-1)\sum_{i=0}^{h-1} t^i$$

$$= 1 + 2\cancel{(t-1)}\frac{t^h - 1}{\cancel{t-1}}$$

$$= 1 + 2t^h - 2$$

$$= 2t^h - 1$$

$$2t^h - 1 \leq n$$

profondità	# nodi	# chiavi
0	1	$2t-1$
1	$2t$	$2t(2t-1)$
2	$(2t)^2$	$(2t)^2(2t-1)$
3	$(2t)^3$	$(2t)^3(2t-1)$
⋮	⋮	⋮
h	$(2t)^h$	$(2t)^h(2t-1)$

$$(2t-1)\sum_{i=0}^{h}(2t)^i$$

$$= \cancel{(2t-1)}\frac{(2t)^{h+1} - 1}{\cancel{2t-1}} = (2t)^{h+1} - 1$$

$$\boxed{2t^h - 1 \le n \le (2t)^{h+1} - 1}$$

b.1

$$2t^h \le n+1$$

$$t^h \le \frac{n+1}{2}$$

$$h \le \log_t \frac{n+1}{2}$$

$$n+1 \le (2t)^{h+1}$$

$$\log_{2t} (n+1) \le h+1$$

$$\log_{2t} \frac{n+1}{2t} \le h$$

$$\boxed{\log_{2t} \frac{n+1}{2t} \le h \le \log_t \frac{n+1}{2}}$$

b.2

$$2t^h - 1 \le n \le (2t)^{h+1} - 1$$

$$2t^h \le n+1$$

$$t^h \le \frac{n+1}{2}$$

$$t \le \sqrt[h]{\frac{n+1}{2}}$$

$$n+1 \le (2t)^{h+1}$$

$$\sqrt[h+1]{n+1} \le 2t$$

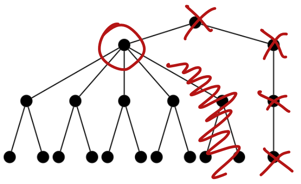
$$\frac{\sqrt[h+1]{n+1}}{2} \le t$$

$$\frac{\sqrt[h+1]{n+1}}{2} \le t \le \sqrt[h]{\frac{n+1}{2}}$$

ESERCIZIO 3

(a) Si enunci e si dimostri un lemma che fornisca una minorazione dei gradi dei figli di ciascun nodo in un heap di Fibonacci.

(b) Si stabilisca se possa esistere un heap di Fibonacci avente la struttura dell'albero a lato.

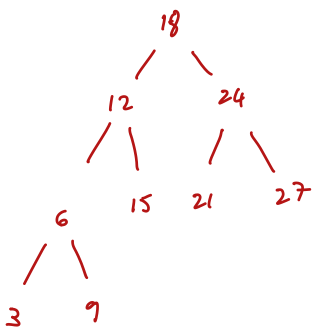


NO!

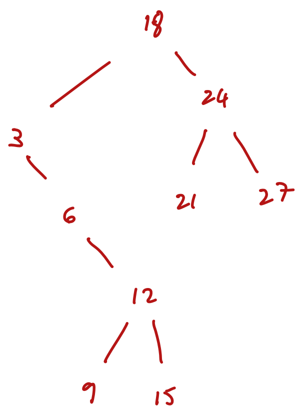
ESERCIZIO 4

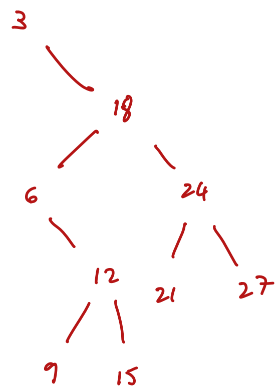
Si descrivano le operazioni di *zig-zag*, *zig-zig* e *zig* in uno splay tree di tipo bottom-up.
Quindi si eseguano nell'ordine dato le seguenti operazioni su uno splay tree la cui configurazione iniziale è quella di un albero binario completo contenente le 9 chiavi $\{3i : 1 \leq i \leq 9\}$:

- Search 3, 9, 21
- Insert 16
- Delete 12
- Search 27

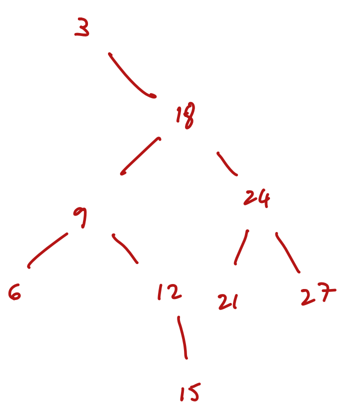


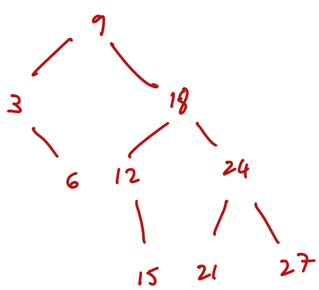
SEARCH 3



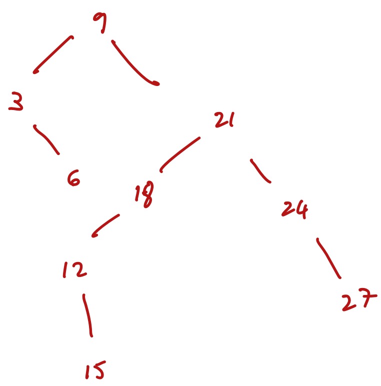


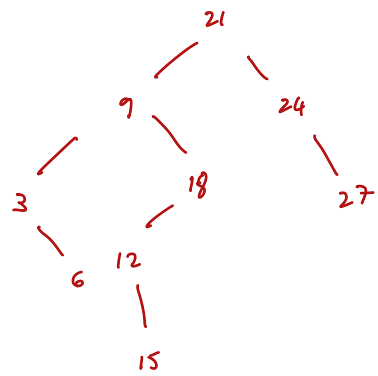
SEARCH 9



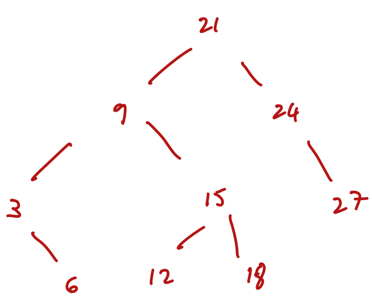


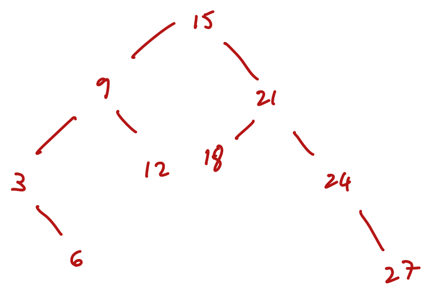
SEARCH 21

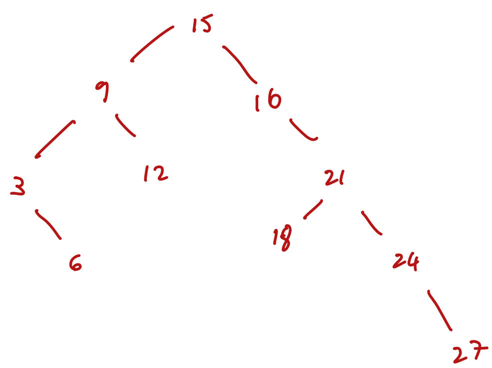


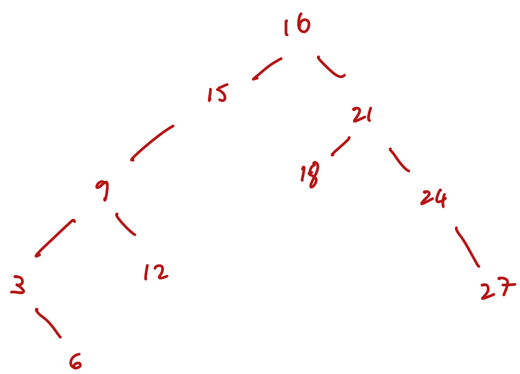


INSERT 16

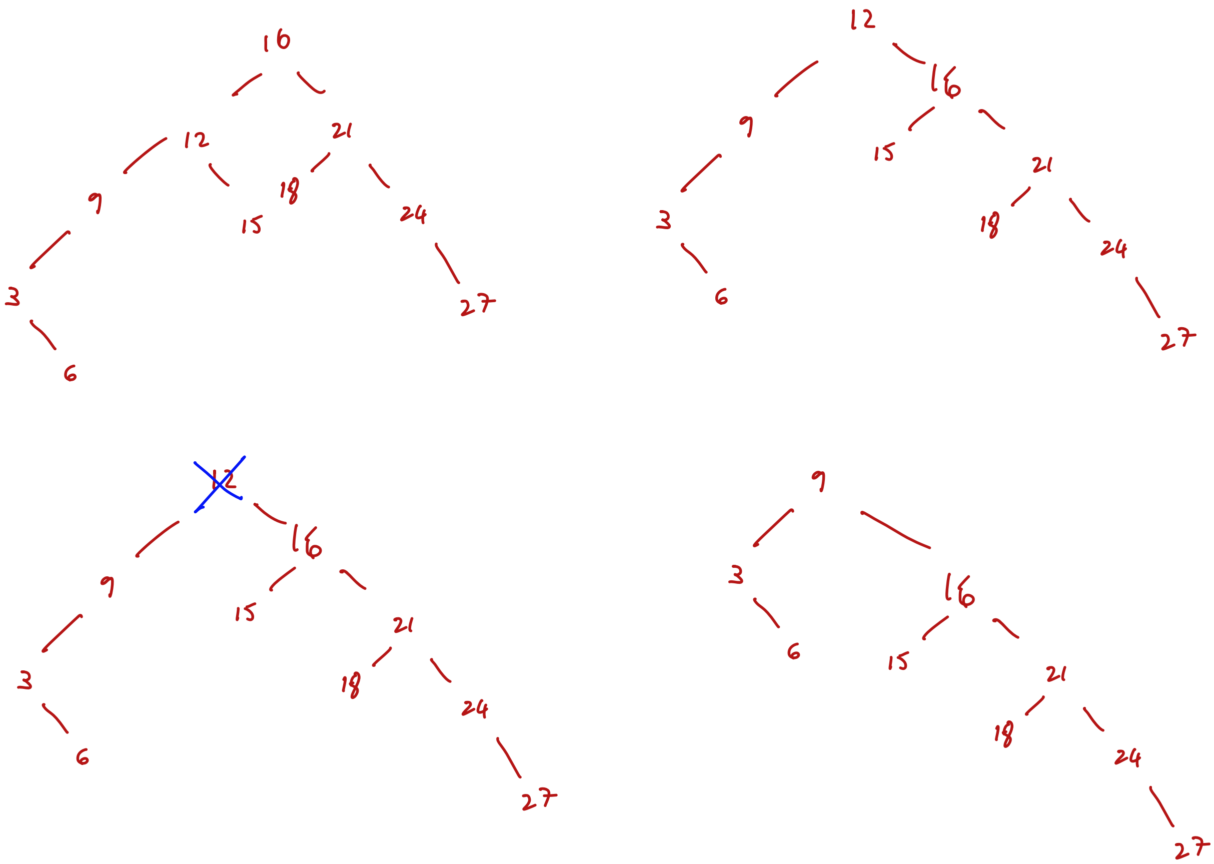








DELETE 12



SEARCH 27