Hvilket af nedenstående svar gælder for følgende rekursionsligning?

$$T(n) = T(n/4) + 1$$

Selected Answer: [None Given]

Answers:

$T(n) = \Theta(\log n)$.

$T(n) = \Theta(n^{1/4})$.

$T(n) = \Theta(n)$.

$T(n) = \Theta(n \log n)$.

$T(n) = \Theta(n^4)$.

Rekursionsligningen kan ikke løses med Master Theorem.

Hvilket af nedenstående svar gælder for følgende rekursionsligning?

$$T(n) = 3T(n/6) + n^{1/2}$$

Selected Answer: [None Given]

Answers:

$T(n) = \Theta(\log n)$.

$T(n) = \Theta(n^{1/2})$.

$T(n) = \Theta(n^{1/2} \log n)$.

$T(n) = \Theta(n^{\alpha})$ med $\alpha = \log_6(3)$.

$T(n) = \Theta(n)$.

$T(n) = \Theta(n \log n)$.

$T(n) = \Theta(n^{\alpha})$ med $\alpha = \log_3(6)$.

$T(n) = \Theta(n^2)$.

Rekursionsligningen kan ikke løses med Master Theorem.

Hvilket af nedenstående svar gælder for følgende rekursionsligning?

$$T(n) = 3T(n/3) + n \log n$$

Selected Answer: [None Given]

Answers:

$T(n) = \Theta(\log n)$.

$T(n) = \Theta(n^{1/3})$.

$T(n) = \Theta(n)$.

$T(n) = \Theta(n \log n)$.

$T(n) = \Theta(n \log^3 n)$.

$T(n) = \Theta(n^3)$.

Rekursionsligningen kan ikke løses med Master Theorem.

Vi ser på en hashtabel H, der bruger linear probing og funktionen $h'(x) = (x + 2) \bmod 7$ som auxiliary hashfunktion. Hashtabellen har allerede nedenstående indhold.

	0	1	2	3	4	5	6
H:	89				23	45	11

Vi indsætter nu værdien 73. Hvilket af nedenstående svar angiver udseendet af H efter indsættelsen?

Selected Answer: [None Given]

Answers:

	0	1	2	3	4	5	6
H:	89			23	45	73	11

	0	1	2	3	4	5	6
H:	89			23	45	11	73

	0	1	2	3	4	5	6
H:	89	73			23	45	11

	0	1	2	3	4	5	6
H:	89		73		23	45	11

	0	1	2	3	4	5	6
H:	89			73	23	45	11

Vi ser på en hashtabel H, der bruger double hashing og funktionerne $h'(x) = (x+2) \bmod 7$ og $h''(x) = (x \bmod 6) + 1$ som auxiliary hashfunktioner. Hashtabellen har allerede nedenstående indhold.

	0	1	2	3	4	5	6
H:	89				23	45	11

Vi indsætter nu værdien 33. Hvilket af nedenstående svar angiver udseendet af H efter indsættelsen?

Selected Answer: [None Given]

Answers:

	0	1	2	3	4	5	6
H:	33	89			23	45	11

	0	1	2	3	4	5	6
H:	89	33			23	45	11

	0	1	2	3	4	5	6
H:	89		33		23	45	11

	0	1	2	3	4	5	6
H:	89			33	23	45	11

Hvor lang tid tager søgning (TREE-SEARCH) i et ubalanceret søgetræ?

Selected Answer: [None Given]

Answers:

$O(1)$

$O(\log n)$

$O(n)$

n 7

1 poi

Hvor lang tid tager søgning (TREE-SEARCH) i et rød-sort træ?

Selected Answer: [None Given]

Answers:

$O(1)$

$O(\log n)$

$O(n)$

1 point

Hvor lang tid tager et inorder gennemløb (INORDER-TREE-WALK) i et ubalanceret søgetræ?

Selected Answer: [None Given]

Answers:

$O(1)$

$O(\log n)$

$O(n)$

Hvor lang tid tager et inorder gennemløb (INORDER-TREE-WALK) i et rød-sort træ?

Selected Answer: [None Given]

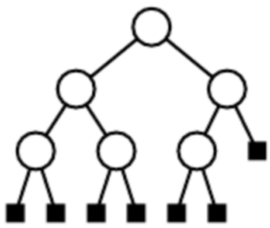
Answers:

$O(1)$

$O(\log n)$

$O(n)$

På hvor mange forskellige måder kan knuderne i nedenstående træ farves, så træet bliver til et lovligt rød-sort træ?



Selected Answer: [None Given]

Answers:

På ingen måder.

På én måde.

På to måder.

På tre måder.

På fire måder.

På fem måder.

Følgende kode har til formål at beregne 2^n.

$$
\begin{array}{l}
\textsc{ToPotens}(n) \\
\quad x = n \\
\quad r = 1 \\
\quad \textbf{while} \;\; x > 0 \\
\qquad r = 2r \\
\qquad x = x - 1 \\
\quad \textbf{return} \;\; r
\end{array}
$$

Er

$$x \geq r$$

en løkke-invariant for algoritmen ToPotens (dvs. er altid sandt, når testen i starten af **while**-løkken udføres) for alle input, der er heltal $n \geq 0$?

Selected Answer: [None Given]

Answers:

Sandt

Falsk

1 point

Er

$$x + r = n + 1$$

en løkke-invariant for algoritmen ToPotens (dvs. er altid sandt, når testen i starten af **while**-løkken udføres) for alle input, der er heltal $n \geq 0$?

Selected Answer: [None Given]

Answers:

Sandt

Falsk

Er

$$x \geq 0$$

en løkke-invariant for algoritmen ToPotens (dvs. er altid sandt, når testen i starten af **while**-løkken udføres) for alle input, der er heltal $n \geq 0$?

Selected Answer: [None Given]

Answers:

Sandt

Falsk

Er

$$r2^x = 2^n$$

en løkke-invariant for algoritmen TOPOTENS (dvs. er altid sandt, når testen i starten af **while**-løkken udføres) for alle input, der er heltal $n \geq 0$?

Selected Answer: [None Given]

Answers:

Sandt

Falsk

For et optimeringsproblem (som her ikke beskrives nærmere) er input givet ved en omkostning (dvs. et tal) c_i for alle heltal i. Det oplyses, at en løsning $l(i)$ til optimeringsproblemet kan beskrives med denne rekursionsligning:

$$l(i) = \begin{cases} 0 & \text{hvis } i = 0 \\ \max\{c_k + l(i-k) \mid 1 \leq k \leq i\} & \text{hvis } i > 0 \text{ og } i \text{ er lige} \\ \min\{c_k + l(i-k) \mid 1 \leq k \leq i\} & \text{hvis } i > 0 \text{ og } i \text{ er ulige} \end{cases}$$

Hvis $l(n)$ findes via dynamisk programmering baseret på ovenstående rekursionsligning, hvilken af nedenstående køretider opnås?

Selected Answer: [None Given]

Answers:

$\Theta(1)$

$\Theta(n)$

$\Theta(n^2)$

$\Theta(n^3)$

Hvis $l(n)$ findes via dynamisk programmering baseret på ovenstående rekursionsligning, hvad er det mindste pladsforbrug, som kan opnås?

Selected Answer: [None Given]

Answers:

$\Theta(1)$

$\Theta(n)$

$\Theta(n^2)$

$\Theta(n^3)$