INSTITUT FOR DATALOGI, AARHUS UNIVERSITET

Det Naturvidenskabelige Fakultet
EKSAMEN
Grundkurser i Datalogi
Algoritmer og Datastrukturer 1 (2003-ordning)
Antal sider i opgavesættet (incl. forsiden): 12 (tolv)
Eksamensdag: Onsdag den 19. marts 2014, kl. 9.00-11.00
Tilladte medbragte hjælpemidler: Alle sædvanlige hjælpemidler (lærebøger og notater). Computer må ikke medbringes.
Materiale der udleveres til eksaminanden:

Årskort ____________________

Navn ____

Skriftlig Eksamen
Algoritmer og Datastrukturer 1 (2003-ordning)

Institut for Datalogi
Aarhus Universitet

Onsdag den 19. marts 2014, kl. 9.00-11.00

Dette eksamenssæt består af en kombination af små skriftlige opgaver og multiple-choice-opgaver. Opgaverne besvares på opgaveformuleringen **som afleveres**.

For hver opgave er angivet opgavens andel af det samlede eksamenssæt.

For multiple-choice-opgaver gælder følgende. Hvert delspørgsmål har præcist et svar. For hvert delspørgsmål, kan du vælge ét svar ved at afkrydse den tilsvarende rubrik. Et multiple-choice-delspørgsmål bedømmes som følgende:

- Hvis du sætter kryds ved det rigtige svar, får du 1 point.
- Hvis du ikke sætter nogen krydser, får du 0 point.
- Hvis du sætter kryds ved et forkert svar, får du $-\frac{1}{k-1}$ point, hvor k er antal svarmuligheder.

For en multiple-choice-opgave med vægt $v\,\%$ og med n delspørgsmål, hvor du opnår samlet s point, beregnes din besvarelse af multiple-choice-opgaven som:

$$\max\left\{0\,,\,\frac{s}{n}\right\}\,\cdot\,v\,\%$$

(Opgavesættet fortsætter)

Opgave 1 (4 %)

	Ja	Nej
n^2 er $O(\frac{1}{2}n^4)$?	☐	☐
$\sqrt{n}$ er $O(\log^7 n)$?	☐	☐
2^n er $O(n^4)$?	☐	☐
$3n^2 + 47$ er $O(60n)$?	☐	☐
$n \log n$ er $O(n^{3/2})$?	☐	☐

Opgave 2 (4 %)

Opskriv følgende funktioner efter stigende orden med hensyn til O-notationen:

$$n \log n$$
$$2^{\frac{1}{2}\log n}$$
$$(\log n)^7$$
$$(3/2)^n$$
$$n + \log n$$

Svar: ____

Opgave 3 (4 %)

Nedenstående spørgsmål vedrører Randomized Quicksort på input af størrelse n.

	$\Theta(n)$	$\Theta(n \log n)$	$\Theta(n^2)$
Worst-case antal kald til Partition proceduren	☐	☐	☐
Forventede antal kald til Partition proceduren	☐	☐	☐
Worst-case tid for Randomized Quicksort	☐	☐	☐
Forventede tid for Randomized Quicksort	☐	☐	☐

Opgave 4 (4 %)

Angiv for hver af nedenstående algoritmer udførselstiden som funktion af n i O-notation.

Algoritme Loop1(n)
  $i = 1$
  while $i \leq n$
    $i = 3 * i$

Algoritme Loop2(n)
  $i = 0$
  while $i \leq n$
    $j = 0$
    while $j \leq i$
      $j = j + 1$
    $i = i + 1$

Algoritme Loop3(n)
  $s = 0$
  $i = 0$
  while $s \leq n$
    $s = s + i$
    $i = i + 1$

Svar Loop1: ______________________

Svar Loop2: ______________________

Svar Loop3: ______________________

Opgave 5 (4 %)

Angiv for hver af nedenstående algoritmer udførselstiden som funktion af n i O-notation.

Algoritme Loop1(n)
  $i = 1$
  while $i \leq n$
    $j = 1$
    while $j \leq i$
      $j = j * 2$
    $i = i * 2$

Algoritme Loop2(n)
  for $i = 0$ **to** n
    $j = 0$
    $s = 0$
    while $s \leq i$
      $j = j + 1$
      $s = s + j$

Algoritme Loop3(n)
  $i = 0$
  $s = 0$
  $q = 0$
  while $q \leq n$
    $i = i + 1$
    $s = s + i$
    $q = q + s$

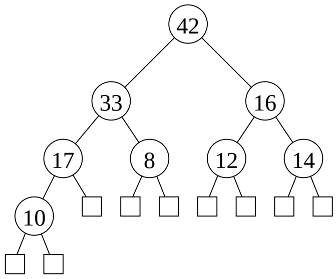
Svar Loop1: ______________________

Svar Loop2: ______________________

Svar Loop3: ______________________

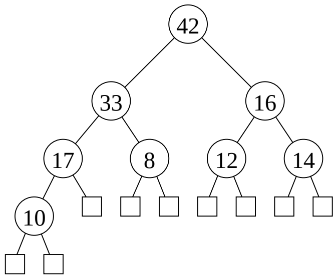
Opgave 6 (4 %)

Tegn hvordan nedenstående binære max-heap ser ud efter indsættelse af elementet 37.



Svar: ______________________________

Tegn hvordan nedenstående binære max-heap ser ud efter en HEAP-EXTRACT-MAX operation.



Svar: ______________________________

Opgave 7 (4 %)

Tegn den binære max-heap efter indsættelse af elementerne 5, 1, 7, 2, 6, 4 og 9 i den givne rækkefølge, startende med den tomme heap.

Svar: ______________________________

Opgave 8 (4 %)

Angiv hvordan nedenstående array ser ud efter anvendelsen af BUILD-MAX-HEAP for arrayet.

1	2	3	4	5	6	7	8	9	10
2	1	4	3	6	5	7	10	8	9

Svar: ____

(Opgavesættet fortsætter)

Opgave 9 (4 %)

Betragt RADIX-SORT anvendt på nedenstående liste af tal ($d = 5$, $k = 5$). Angiv den delvist sorterede liste efter at radix-sort har sorteret tallene efter de *tre* mindst betydende cifre.

21224 54123 43123 10224 32123

Svar: ______________________________

Opgave 10 (4 %)

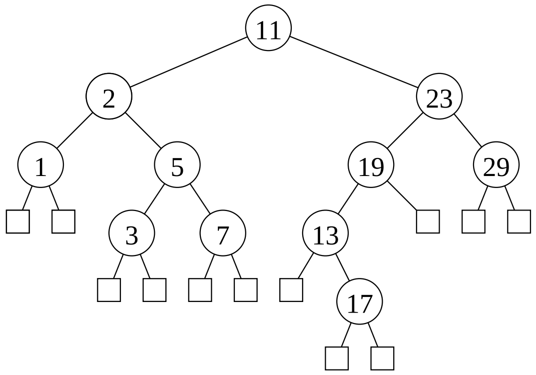
Angiv resultatet af at anvende PARTITION(A,2,14) på nedenstående array.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
A	1	16	2	15	3	14	4	13	5	12	6	11	7	9	10

Svar: ______________________________

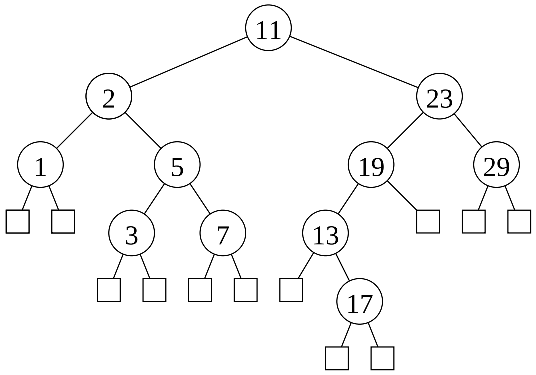
Opgave 11 (4 %)

Tegn hvordan nedenstående ubalancerede binære søgetræ ser ud efter indsættelse af elementet 12.



Svar: ______________________________

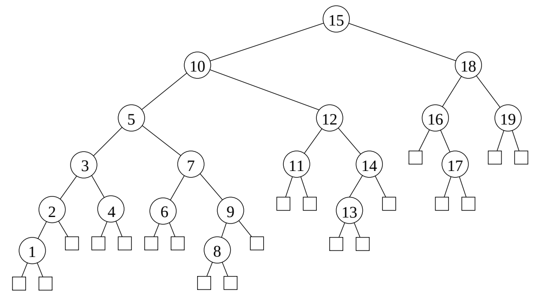
Tegn hvordan nedenstående ubalancerede binære søgetræ ser ud efter slettelse af elementet 11.



Svar: ______________________________

Opgave 12 (4 %)

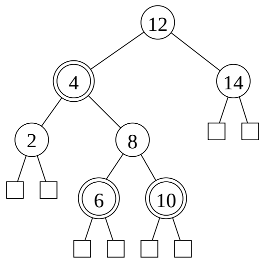
Angiv hvorledes knuderne i nedenstående binære søgetræ kan farves røde og sorte, således at det resulterende træ er et lovligt rød-sort træ.



Svar: ________________________________

Opgave 13 (4 %)

Tegn hvordan nedenstående rød-sorte træ (dobbeltcirkler angiver røde knuder) ser ud efter indsættelse af elementet 5.



Svar: ________________________________

Opgave 14 (4 %)

Tegn en hashtabel hvor der anvendes kædede lister til at håndtere kollisioner, når hashfunktionen er $h(k) = 5k \bmod 7$ og der indsættes elementerne 3, 1, 4, 8, 6, 11, og 2 i den givne rækkefølge.

Svar: ____

Opgave 15 (4 %)

Tegn hvordan en hashtabel der anvender *linear probing* ser ud efter at elementerne 9, 7, 2, 5, og 3 indsættes i den givne rækkefølge, når hashfunktionen er $h(k) = 3k \bmod 7$.

0	1	2	3	4	5	6

Svar: ____

Opgave 16 (4 %)

Tegn hvordan en hashtabel der anvender *dobbelt hashing* ser ud efter at elementerne 7, 1, 2, 0, og 11 indsættes i den givne rækkefølge, når hashfunktionerne er $h_1(k) = 3k \bmod 11$, $h_2(k) = 1 + (2k \bmod 5)$, og hastabellen har størrelse 7.

0	1	2	3	4	5	6

Svar: ____

Opgave 17 (4 %)

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering. Angiv for hver knude rangen af knuden.

makeset(a)
makeset(b)
makeset(c)
makeset(d)
makeset(e)
makeset(f)
union(a,b)
union(c,d)
union(a,c)
union(b,e)
union(a,f)

Svar: ______________________________

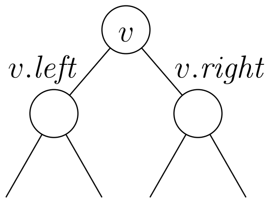
Opgave 18 (4 %)

Betragt en liste af n par $(x_1, w_1), \ldots, (x_n, w_n)$, hvor x_i og w_i er reelle tal og $x_1 < x_2 < \cdots < x_n$. For $1 \leq k \leq n$ udgør $(x_1, w_1), \ldots, (x_k, w_k)$ et præfiks af listen, og vi definerer vægten af præfikset til at være $w_1 + \cdots + w_k$. Vi ønsker at vedligeholde den maksimale præfiksvægt for listen, dvs. maksimum taget over alle præfiksernes vægt.

F.eks. har listen $(1, 4), (2, -3), (5, 8), (7, -2)$ maksimal præfiksvægt $4 + (-3) + 8 = 9$.

Betragt et søgetræ hvor hver knude v gemmer et par $v.x$ og $v.w$, og knuderne er ordnet venstre-mod-højre efter stigende $v.x$. Derudover gemmes i v også $v.\mathit{sum}$ og $v.\mathit{maxp}$, som er hhv. summen af alle vægtene i v's undertræ, og den maksimale præfiksvægt af dellisten gemt i v's undertræ.

Angiv hvorledes disse værdier kan beregnes når den tilsvarende information er kendt ved de to børn $v.\mathit{left}$ og $v.\mathit{right}$ (det kan antages at disse begge eksisterer).

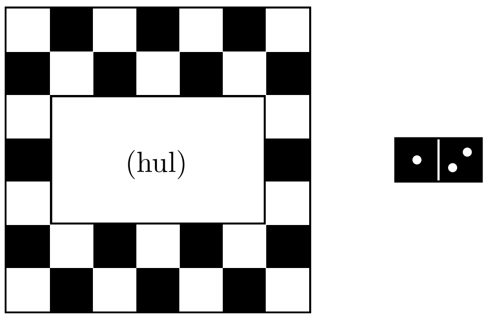


Svar $v.\mathit{sum}$ = ____________________

Svar $v.\mathit{maxp}$ = ____________________

Opgave 19 (4 %)

Betragt højrestående 7×7 skakbræt med et 3×5 hul. Vi ønsker at dække de resterende 34 felter med dominobrikker.



For at bevise at skakbrættet ikke kan dækkes med dominobrikker, betragter vi følgende transitionssystem, der indfanger antallet af ikke-dækkede sorte (s) og hvide (h) felter:

$$[s, h] \triangleright [s - 1, h - 1]$$

Angiv en invariant, hvorfra det følger at de 34 felter ikke kan dækkes med dominobrikker.

Invariant: ____________________

Transitionssystem GCD
Konfigurationer: $\{[m,n] \mid \text{heltal } m,n \wedge m \geq 0 \wedge n \geq 0 \wedge m{+}n \geq 1\}$

$[m,n] \triangleright [n,m]$ **if** $m < n$
$[m,n] \triangleright [m-n,n]$ **if** $m \geq n \wedge n > 0$

Opgave 20 (4 %)

For hvert af nedenstående udsagn, angiv om de er en invariant for ovenstående transitionssystem GCD. Startkonfigurationen antages at være $[n_0, m_0]$, hvor $n_0 \geq 1$ og $m_0 \geq 1$. I det følgende betegner $\gcd(x,y)$ den største fælles divisor i to heltal x og y.

	Ja	Nej
$n \neq 0$	☐	☐
$\gcd(m,n) = \gcd(m_0,n_0)$	☐	☐
$m + n = m_0 + n_0$	☐	☐
$m + n \geq 1$	☐	☐
$m \cdot n \geq \gcd(m_0,n_0)$	☐	☐

Opgave 21 (4 %)

For hver af nedenstående funktioner, angiv om de er en termineringsfunktion for ovenstående transitionssystem GCD.

	Ja	Nej
$\mu(m,n) = m + n$	☐	☐
$\mu(m,n) = 2m + n$	☐	☐
$\mu(m,n) = m + 2n$	☐	☐
$\mu(m,n) = m$	☐	☐
$\mu(m,n) = 2n - m$	☐	☐

Hvis et array $A[1..n]$ indeholder et element $A[j]$, der forekommer flere end $n/2$ gange i A, så siges $A[j]$ at være et *majoritetselement*.

Algoritme Majoritet($A[1..n]$)
Inputbetingelse : Array $A[1..n]$ med n heltal, hvor $n \geq 1$ og ét tal forekommer $> n/2$ gange i A
Outputkrav : j, hvor $A[j]$ er majoritetselementet
Metode : $i \leftarrow 1$;
$j \leftarrow 1$;
$c \leftarrow 1$;
$\{I\}$ **while** $i < n$ **do**
  $i \leftarrow i + 1$;
  if $c = 0$ **then**
    $j \leftarrow i$;
    $c \leftarrow 1$
  else if $A[i] = A[j]$ **then**
    $c \leftarrow c + 1$
  else
    $c \leftarrow c - 1$

Opgave 22 (4 %)

For hvert af nedenstående udsagn, angiv om de er en invariant I for ovenstående algoritme Majoritet. For $1 \leq k \leq i$, lad $\text{count}(k, i)$ betegne antal forekomster af $A[k]$ i $A[1..i]$, dvs. $\text{count}(k, i) = |\{\ell \mid 1 \leq \ell \leq i \ \wedge\ A[\ell] = A[k]\}|$.

	Ja	Nej
$c = \text{count}(j, i)$	☐	☐
$c \leq \text{count}(j, i)$	☐	☐
$c = i - j$	☐	☐
$2 \cdot \text{count}(k, i) + c \leq i$, for $A[k] \neq A[j]$	☐	☐
$2 \cdot \text{count}(j, i) - c \leq i$	☐	☐

Opgave 23 (4 %)

For hver af nedenstående funktioner, angiv om de er en termineringsfunktion for ovenstående algoritme Majoritet.

	Ja	Nej
$\mu(n, i, j, c) = i$	☐	☐
$\mu(n, i, j, c) = n - i - c$	☐	☐
$\mu(n, i, j, c) = n - i$	☐	☐
$\mu(n, i, j, c) = i - j$	☐	☐
$\mu(n, i, j, c) = 2(n - i) - c$	☐	☐

(Opgavesættet fortsætter)

Opgave 24 (4 %)

Nedenstående algoritme er en variation af binær søgning efter et element x i et sorteret array $A[1..n]$, hvor man ikke nødvendigvis vælger det midterste element i et delarray $A[L..H]$ at sammenligne x med, men vælger et $A[m]$ afhængig om værdien af x ligger tættest på $A[L]$ eller $A[H]$. For at vise gyldigheden af algoritmen skal $I_{L,H}$ og I_A være invarianter omkring variablerne L og H, og sammenhængen med A. Angiv invarianter hvormed gyldigheden af algoritmen kan bevises (bevis for invarianterne kræves ikke).

Algoritme InterpolationSearch$(A[1..n], x)$
Inputbetingelse : Sorteret array $A[1..n]$ med n heltal og et heltal x, hvor $n \geq 1$
Outputkrav : r, hvor $r = -1$ hvis $x \notin A$, ellers $A[r] = x$
Metode : $r \leftarrow -1$;
$L \leftarrow 1$;
$H \leftarrow n$;
$\{I\}$ **while** $r = -1 \ \wedge \ L \leq H$ **do**
$\quad m \leftarrow \min\left\{H, \max\left\{L, L + \left\lfloor \frac{(x-A[L])(H-L)}{A[H]-A[L]} \right\rfloor \right\}\right\}$;
$\quad$ **if** $x = A[m]$ **then**
$\quad\quad r \leftarrow m$
$\quad$ **else if** $x < A[m]$ **then**
$\quad\quad H \leftarrow m - 1$
$\quad$ **else**
$\quad\quad L \leftarrow m + 1$

Svar $I_{L,H}$: ________________________________

Svar I_A: ________________________________

For at kunne bevise at algoritmen terminerer, kræves en passende termineringsfunktion. Angiv en termineringsfunktion (bevis for at termineringsfunktionen har de nødvendige egenskaber kræves ikke).

Svar μ: ________________________________

Opgave 25 (4 %)

Antag at en binær max-heap med n elementer er gemt i de første n indgange i et array af størrelsen N. Operationerne Insert og Extract-Max implementeres som for en standard binær max-heap, på nær Insert når $n = N$, hvor heapen først kopieres over i et nyt array af dobbelt størrelse $2N$, dvs. N fordobles, hvorefter Insert operationen udføres som for en standard binær max-heap. Med en passende potentialefunktion kan man argumentere for at Insert tager amortiseret $O(\log n)$ tid og Extract-Max tager amortiseret $O(1)$ tid. Angiv en sådan potentialefunktion.

Svar $\Phi =$ ________________________________