INSTITUT FOR DATALOGI, AARHUS UNIVERSITET

Science and Technology
EKSAMEN
Grundkurser i Datalogi
Algoritmer og Datastrukturer 1 (2003-ordning)
Antal sider i opgavesættet (incl. forsiden): 12
Eksamensdag: Mandag den 4. april 2016, 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

Mandag den 4. april 2016, kl. 9.00-11.00

Dette eksamenssæt består af en mængde multiple-choice-opgaver. Opgaverne besvares på opgaveformuleringen **som afleveres**.

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

Hvert delspørgsmål har præcist et rigtigt svar. For hvert delspørgsmål, kan du vælge **max ét svar** ved at afkrydse den tilsvarende rubrik. Et 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 opgave med vægt $v\,\%$ og med n delspørgsmål, hvor du opnår samlet s point, beregnes din besvarelse af opgaven som:

$$\frac{s}{n} \cdot v\,\%$$

Bemærk at det er muligt at få negative point for en opgave.

Opgave 1 (10 %)

I det følgende angiver $\log n$ 2-tals-logaritmen af n.

	Ja	Nej
$7n$ er $O(2n)$?	☐	☐
$2n + 5n$ er $O(n + \frac{5}{n})$?	☐	☐
2^n er $O(n^2)$?	☐	☐
$n^2\sqrt{n}$ er $O(n^3)$?	☐	☐
$\log n$ er $O(\sqrt{n})$?	☐	☐
$n^2 + \sqrt{n}$ er $O(n^{2.5})$?	☐	☐
$2^{(n^2)}$ er $O((2^n)^2)$?	☐	☐
$2^{\log n}$ er $O(\sqrt{n})$?	☐	☐
$n^{2/3}$ er $O(\frac{2}{3}n)$?	☐	☐
$2^{\log n}$ er $O((\log n)^2)$?	☐	☐
$7^2 + 2^7$ er $O(7)$?	☐	☐
$\log n + \sqrt{n}$ er $O(\log n)$?	☐	☐
$n \cdot \log n$ er $O(n^2)$?	☐	☐
$\log n$ er $O(n)$?	☐	☐
n^7/n^3 er $O(n^3)$?	☐	☐
3^n er $O(3 \cdot 2^n)$?	☐	☐
$1/n$ er $O(1)$?	☐	☐
4^n er $O(n^n)$?	☐	☐
n^n er $O(n!)$?	☐	☐
$n^4 + n^3$ er $\Theta(7n^4)$?	☐	☐

Opgave 2 (4 %)

Betragt en $n \times n$ matrix M af heltal, hvor alle rækker og søjler er voksende, dvs. $M[i,j] \leq M[i',j']$ for alle $1 \leq i \leq i' \leq n$ og $1 \leq j \leq j' \leq n$.

Hvad er det bedste worst-case tid man kan opnå for at søge efter et heltal i M?

$O(\log n)$	$O(n)$	$O(n \log n)$	$O(n^2)$	$O(n\sqrt{n})$	$O(\sqrt{n})$	$O((\log n)^2)$
☐	☐	☐	☐	☐	☐	☐

Opgave 3 (10 %)

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

```
Algoritme Loop1(n)
  s = 0
  for i = 1 to n
    s = s + i
```

```
Algoritme Loop2(n)
  i = 1
  while i ≤ n
    j = n
    while j > 0
      j = j − 1
    i = i ∗ 2
```

```
Algoritme Loop3(n)
  i = 0
  while i ≤ n
    j = i
    while j > 0
      j = ⌊j/2⌋
    i = i + 1
```

```
Algoritme Loop4(n)
  i = 1
  while i ≤ n
    j = 1
    while j ≤ i
      j = j ∗ 2
    i = i ∗ 2
```

```
Algoritme Loop5(n)
  i = 1
  s = 0
  while s ≤ n
    j = 1
    while j ≤ i
      j = j + 1
    s = s + i
    i = i + 1
```

```
Algoritme Loop6(n)
  i = 1
  j = 1
  s = 0
  while s ≤ n
    while j ≤ s
      j = 2 ∗ j
    s = s + i
    i = i + 1
```

	$O(\log n)$	$O(n)$	$O(n \log n)$	$O(n^2)$	$O(n\sqrt{n})$	$O(\sqrt{n})$	$O(n^3)$	$O((\log n)^2)$
Loop1	☐	☐	☐	☐	☐	☐	☐	☐
Loop2	☐	☐	☐	☐	☐	☐	☐	☐
Loop3	☐	☐	☐	☐	☐	☐	☐	☐
Loop4	☐	☐	☐	☐	☐	☐	☐	☐
Loop5	☐	☐	☐	☐	☐	☐	☐	☐
Loop6	☐	☐	☐	☐	☐	☐	☐	☐

Opgave 4 (4 %)

1	2	3	4	5	6	7	8	9	10	11
13	11	5	10	7	1	2	4	8	6	3

Angiv hvordan ovenstående binære max-heap ser ud efter Heap-Extract-Max.

1	2	3	4	5	6	7	8	9	10	
11	7	5	10	3	1	2	4	8	6	☐

1	2	3	4	5	6	7	8	9	10	
11	5	10	7	1	2	4	8	6	3	☐

1	2	3	4	5	6	7	8	9	10	
11	10	5	8	7	1	2	4	3	6	☐

1	2	3	4	5	6	7	8	9	10	
11	10	8	7	6	5	4	3	2	1	☐

Opgave 5 (4 %)

Angiv den binære max-heap efter indsættelse af elementerne 1, 3, 2, 5, 4, 6 og 7 i den givne rækkefølge, startende med den tomme heap.

1	2	3	4	5	6	7	
7	4	6	3	1	2	5	☐

1	2	3	4	5	6	7	
7	6	4	5	2	3	1	☐

1	2	3	4	5	6	7	
7	4	6	1	3	5	2	☐

1	2	3	4	5	6	7	
7	4	6	1	3	2	5	☐

Opgave 6 (4 %)

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

Angiv hvordan ovenstående array ser ud efter anvendelsen af Build-Max-Heap.

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

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

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

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

Opgave 7 (4 %)

Betragt Radix-Sort anvendt på nedenstående liste af tal ($d = 5$, $k = 2$).

20020 11111 20120 10020 10111

Angiv den delvist sorterede liste efter at radix-sort har sorteret tallene efter de <u>*tre*</u> mindst betydende cifre.

11111	10111	20020	20120	10020	☐
10020	20020	10111	11111	20120	☐
10020	10111	11111	20020	20120	☐
20020	10020	11111	10111	20120	☐
11111	10111	20120	20020	10020	☐

Opgave 8 (4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14
4	27	5	10	12	25	14	7	42	24	2	13	99	1

Angiv resultatet af at anvende PARTITION(A,2,12) på ovenstående array.

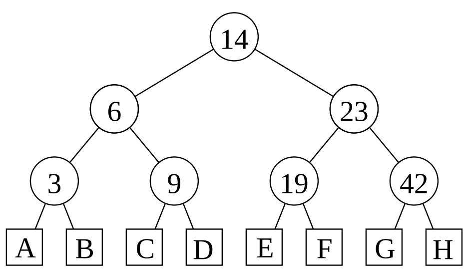
1	2	3	4	5	6	7	8	9	10	11	12	13	14	
4	5	10	12	7	2	13	27	25	14	42	24	99	1	☐

1	2	3	4	5	6	7	8	9	10	11	12	13	14	
4	5	10	12	7	2	13	27	42	24	25	14	99	1	☐

1	2	3	4	5	6	7	8	9	10	11	12	13	14	
4	5	10	12	7	2	13	27	42	25	24	14	99	1	☐

1	2	3	4	5	6	7	8	9	10	11	12	13	14	
2	1	4	5	10	12	7	27	25	14	42	24	13	99	☐

Opgave 9 (4 %)

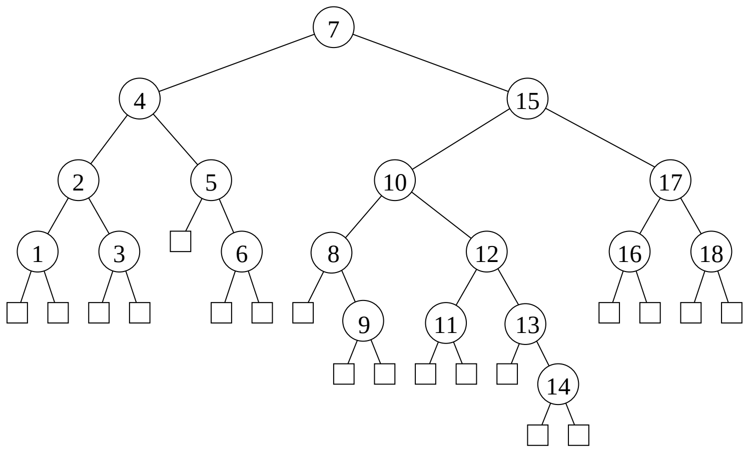


Angiv i hvilke blade A-H i ovenstående ubalancerede binære søgetræ elementerne 11, 2, 8, 21 og 7 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående syv elementer).

	A	B	C	D	E	F	G	H
`Insert(11)`	☐	☐	☐	☐	☐	☐	☐	☐
`Insert(2)`	☐	☐	☐	☐	☐	☐	☐	☐
`Insert(8)`	☐	☐	☐	☐	☐	☐	☐	☐
`Insert(21)`	☐	☐	☐	☐	☐	☐	☐	☐
`Insert(7)`	☐	☐	☐	☐	☐	☐	☐	☐

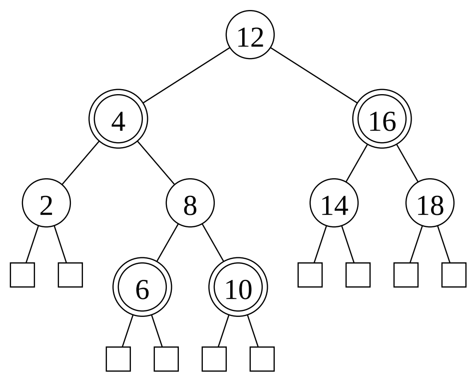
Opgave 10 (4 %)

For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde

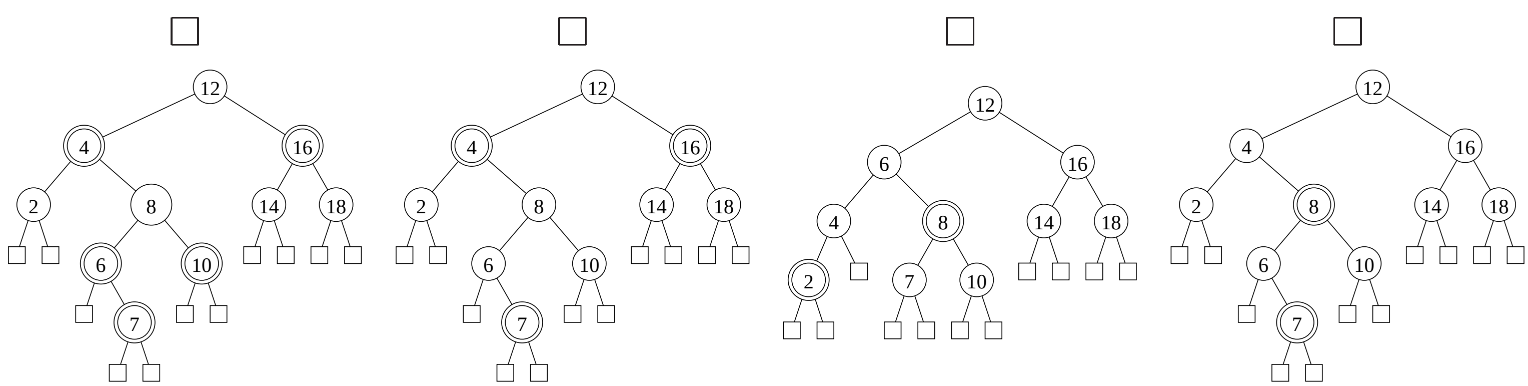


	Ja	Nej
$2, 5, 8, 12, 14, 15$	☐	☐
$2, 6, 9, 12, 14, 15$	☐	☐
$10, 14$	☐	☐
$2, 5, 9, 10, 13, 17$	☐	☐
$1, 3, 6, 9, 12, 14, 15$	☐	☐

Opgave 11 (4 %)



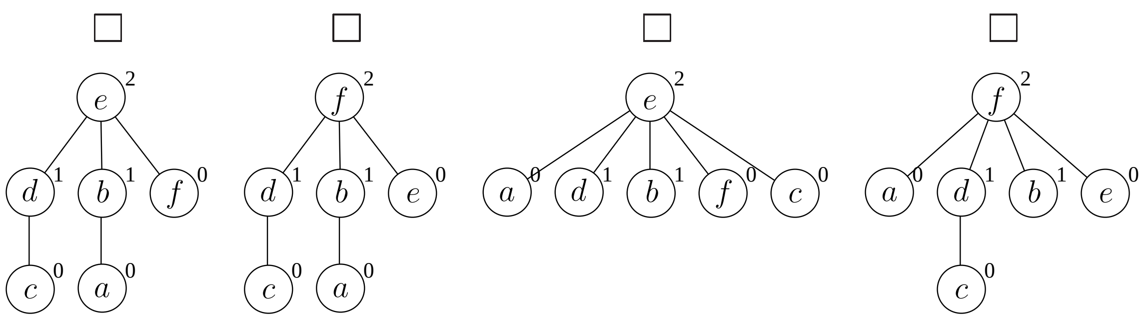
Angiv det resulterende rød-sorte træ når man indsætter 7 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).



Opgave 12 (4 %)

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

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



Opgave 13 (4 %)

I følgende hashtabel er anvendt *linear probing* med hashfunktionen $h(k) = (3k + 2) \bmod 7$.

0	1	2	3	4	5	6
4		0				13

Angiv positionerne de tre elementer 2, 6 og 7 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 0, 4 og 13).

	0	1	2	3	4	5	6
`Insert(2)`	☐	☐	☐	☐	☐	☐	☐
`Insert(6)`	☐	☐	☐	☐	☐	☐	☐
`Insert(7)`	☐	☐	☐	☐	☐	☐	☐

Opgave 14 (4 %)

I følgende hashtabel af størrelse 11 er anvendt *kravdratisk probing* med hashfunktionen $h(k,i) = (2k + 3i + i^2) \bmod 11$.

0	1	2	3	4	5	6	7	8	9	10
				13				4		16

Angiv positionerne de tre elementer 7, 5 og 2 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 4, 13 og 16).

	0	1	2	3	4	5	6	7	8	9	10
`Insert`(7)	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
`Insert`(5)	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
`Insert`(2)	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

Opgave 15 (4 %)

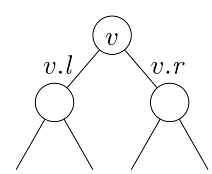
Betragt en liste af n punkter $(x_1, y_1), \ldots, (x_n, y_n)$, hvor x_i og y_i er reelle tal og $x_1 < x_2 < \cdots < x_n$. Vi ønsker at finde en sammenhængende række af punkter

$$(x_i, y_i), (x_{i+1}, y_{i+1}), \ldots, (x_{j-1}, y_{j-1}), (x_j, y_j),$$

hvor $i \leq j$, således at $Y_{i,j} = \sum_{k=i}^{j} y_k = y_i + y_{i+1} + \cdots + y_{j-1} + y_j$ er størst mulig. Vi ønsker for en dynamisk liste af punkter at vedligeholde denne maksimale sammenhængende y-sum, *maxysum*.

Betragt et søgetræ hvor hver knude v gemmer et punkt $(v.x, v.y)$, og knuderne er ordnet venstre-mod-højre efter stigende $v.x$. Derudover gemmes i en knude v, der i undertræet indeholer punkterne $(x_k, y_k), \ldots, (x_\ell, y_\ell)$ fire værdier $v.sum = Y_{k,\ell}$, $v.maxysum = \max_{k \leq i \leq j \leq \ell} Y_{i,j}$, $v.pre = \max\{0, \max_{k \leq j \leq \ell} Y_{k,j}\}$, og $v.suf = \max\{0, \max_{k \leq i \leq \ell} Y_{i,\ell}\}$.

Angiv hvorledes $v.sum$ og $v.maxysum$ kan beregnes når den tilsvarende information (incl. *pre* og *suf*) er kendt ved de to børn $v.l$ og $v.r$ (det kan antages at disse begge eksisterer).



$$v.sum = \begin{cases} v.l.sum + v.r.sum & \square \\ v.l.sum + v.y + v.r.sum & \square \\ v.l.suf + v.r.pre & \square \\ v.l.suf + v.y + v.r.pre & \square \end{cases}$$

$$v.maxysum = \begin{cases} \max\{v.l.maxysum, v.r.maxysum\} & \square \\ \max\{v.l.maxysum, v.y, v.r.maxysum\} & \square \\ \max\{v.l.maxysum, v.l.suf + v.y + v.r.pre, v.r.maxysum\} & \square \\ \max\{v.l.maxysum, v.l.suf + v.r.pre, v.r.maxysum\} & \square \end{cases}$$

Transitionssystem DIVIDE
Konfigurationer: $\{[x, y, q, r, p] \mid \text{heltal } x, q, r \geq 0 \text{ og } y, p \geq 1\}$

$[x, y, q, r, p] \triangleright [x, y, q, r, \lfloor p/2 \rfloor]$ **if** $p \geq 2$
$[x, y, q, r, p] \triangleright [x - yp, y, q + p, r, p]$ **if** $x \geq yp$
$[x, y, q, r, p] \triangleright [0, y, q, r + x, p]$ **if** $0 < x < y$

Opgave 16 (4 %)

For hvert af nedenstående udsagn, angiv om de er en invariant for ovenstående transitionssystem DIVIDE. Startkonfigurationen antages at være $[x_0, y_0, 0, 0, x_0]$, hvor $x_0 \geq 0$ og $y_0 \geq 1$.

	Ja	Nej
$x_0 = qy + r$	☐	☐
$x_0 = x + qy + r$	☐	☐
$x_0 = x - (qy + r)$	☐	☐
$x \leq p$	☐	☐
$p \geq 1$	☐	☐

Opgave 17 (4 %)

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

	Ja	Nej
$\mu(x, y, q, r, p) = x$	☐	☐
$\mu(x, y, q, r, p) = x + p$	☐	☐
$\mu(x, y, q, r, p) = x + r$	☐	☐
$\mu(x, y, q, r, p) = x - yp$	☐	☐
$\mu(x, y, q, r, p) = qy + r$	☐	☐

Opgave 18 (4 %)

Betragt en kø implementeret i et cyklisk array Q, hvor $Q.head$ angiver hovedet af køen og $Q.tail$ er den næste ledige plads i køen (som beskrevet i [CLRS]).

Hvilke af følgende funktioner beregner korrekt størrelsen af køen?

	Ja	Nej
$Q.tail - Q.head$	☐	☐
$\lvert Q.tail - Q.head \rvert$	☐	☐
$\lvert Q.tail - Q.head \rvert \bmod Q.length$	☐	☐
$(Q.tail - Q.head + Q.length - 1) \bmod Q.length$	☐	☐
$(Q.tail - Q.head + Q.length) \bmod Q.length$	☐	☐

Antag at et sorteret array $A[1..n]$ indeholder $n \geq 2$ forskellige heltal, dvs. $A[1] < A[2] < \cdots < A[n-1] < A[n]$. Givet et positivt heltal $x > 0$, identificerer nedenstående algoritme om der findes $1 \leq i < j \leq n$, hvor $x = A[j] - A[i]$.

Algoritme FINDDIFF$(A[1..n], x)$
Inputbetingelse : Sorteret array $A[1..n]$ med n forskellige heltal, heltal $x > 0$
Outputkrav : $1 \leq i < j \leq n$, hvor $x = A[j] - A[i]$;
ellers $j > n$ hvis intet sådant par findes
Metode : $i \leftarrow 1$;
$j \leftarrow 1$;
$\{I\}$ **while** $j \leq n$ **and** $x \neq A[j] - A[i]$ **do**
  if $x > A[j] - A[i]$ **then**
    $j \leftarrow j + 1$
  else
    $i \leftarrow i + 1$

Opgave 19 (4 %)

For hvert af nedenstående udsagn, angiv om de er en invariant I for ovenstående algoritme FINDDIFF.

	Ja	Nej
$i \leq j$	☐	☐
$1 \leq i \leq j \leq n$	☐	☐
$x \neq A[j] - A[i]$	☐	☐
For alle $1 \leq i' < j' \leq n : (i' \geq i \ \vee \ j' \geq j) \Rightarrow x \neq A[j'] - A[i']$	☐	☐
For alle $1 \leq i' < j' \leq n : (i' < i \ \vee \ j' < j) \Rightarrow x \neq A[j'] - A[i']$	☐	☐

Opgave 20 (4 %)

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

	Ja	Nej
$\mu(i, j, n) = i + j$	☐	☐
$\mu(i, j, n) = j - i$	☐	☐
$\mu(i, j, n) = 2n - i - j$	☐	☐
$\mu(i, j, n) = 2n + 1 - i - j$	☐	☐
$\mu(i, j, n) = 3n + 2(j - i)$	☐	☐

Opgave 21 (4 %)

Lad $\|x\|$ betegne antal bit med værdien 1 i den binære repræsentation af et ikke-negativt heltal x, f.eks. $\|14_{10}\| = \|1110_2\| = 3$. Nedenstående algoritme beregner $\|x\|$.

Algoritme BITS(x)
Inputbetingelse : Heltal $x \geq 0$
Outputkrav : $r = \|x\|$
Metode : $r \leftarrow 0$;
$\{I\}$ **while** $x > 0$ **do**
  if x ulige **then**
    $x \leftarrow x - 1$;
    $r \leftarrow r + 1$
  $x \leftarrow x/2$

For hvert af nedenstående udsagn, angiv om de er en invariant I for ovenstående algoritme BITS, hvor x_0 angiver den initielle værdi af x.

	Ja	Nej
$r = \|x\|$	☐	☐
$r + \|x\| = \|x_0\|$	☐	☐
$r + \|x_0\| = \|x\|$	☐	☐
$r + \|x\| \leq x_0$	☐	☐
$\|x\| + 2^r = x_0$	☐	☐

Opgave 22 (4 %)

Betragt en uordnet liste af n heltal, hvor vi i $O(1)$ tid kan indsætte et nyt heltal, og i $O(n)$ tid kan udføre operationen NegateRemove, som fjerner alle ikke-positive tal ($x \leq 0$) fra listen og erstatter ethvert positivt heltal $y > 0$ med det tilsvarende negative heltal $-y$. F.eks. NegateRemove$(3, -4, -2, 7, 6, -2) = (-3, -7, -6)$.

Med en passende potentialefunktion kan man argumentere for at både indsættelser og NegateRemove tager amortiseret $O(1)$ tid. Angiv for hver af nedenstående om dette er en sådan potentialefunktion Φ, hvor P er antal positive tal i listen og N er antal ikke-positive tal i listen.

	Ja	Nej
N	☐	☐
P	☐	☐
$2N + P$	☐	☐
$N + P$	☐	☐
$N + 2P$	☐	☐