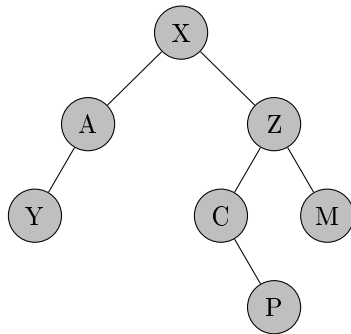


Uppgift 5, Träd (xp)

Uppgift 5.1 Traversering av binärt träd



Illustrationen till vänster visar ett träd. Betrakta trädet som ett **binärt** träd. Nedanstående frågor handlar om traversering av trädet. I svaren, skriv **nodernas etiketter, separerade med kommatecken**, t.ex. "A,B,C,D,E,F,G".

- I vilken ordning kommer noderna att besökas om trädet traverseras enligt **bredden-först**? (X,A,Z,Y,C,M,P)
- I vilken ordning kommer noderna att besökas om trädet traverseras enligt **djupet-först, pre-order**? (X,A,Y,Z,C,P,M)
- I vilken ordning kommer noderna att besökas om trädet traverseras enligt **djupet-först, in-order**? (Y,A,X,C,P,Z,M)
- I vilken ordning kommer noderna att besökas om trädet traverseras enligt **djupet-först, post-order**? (Y,A,P,C,M,Z,A)

Uppgift 5.2 Traversering av ordnat träd

Illustrationen till vänster är densamma som för föregående uppgift. Ange nedan för var och en av traverseringsordningarna om dom skulle ge *samma* nodsekvens eller en *förändrad* nodsekvens om vi betraktade trädet som ett **ordnat** träd i stället för ett **binärt** träd.

	Samma	Förändrad
Bredden-först	X	
Djupet-först, preorder	X	
Djupet-först, inorder		X
Djupet-först, postorder	X	

- Det är bara Djupet-först-inorder som skiljer mellan ett binärt träd och ett ordnat träd. Ett ordnat träd skiljer inte på "vänsterbarn" och "högerbarn", så en nod (X) utan vänsterbarn men med ett högerbarn (Y) kommer att traverseras X-Y om trädet betraktas som binärt men Y-X om trädet betraktas som ordnat.
- Nu hade jag klipp-och-klistrat in en tabbe i svaren, då två av alternativen var samma. Jag har rättat om uppgiften manuellt och justerat poängen.

Uppgift 7

Uppgift 7.1 Ordning

Vilka av följande datatyper har följande egenskaper?

	Oordnad	Linjärt ordnad	Hierarkiskt ordnad	Inget av de övriga alternativen
Graf	X			
Hög			X	
Binärt sökträd			X	
Lista		X		
Stack		X		
Heltal				X

Uppgift 7.2 Sortering

	Sorterad	Partiellt sorterad	Inget
Graf			X
Hög		X	
Binärt sökträd	X		
Lista			X
Stack			X
Heltal			X

Kommentarer:

- Alla träd är hierarkiskt ordnade.
- En GRAF är OORDNAD och är därför VARKEN sorterad eller osorterad.
- En HÖG är ett PARTIELLT SORTERAT binärt TRÄD och därmed HIERARKISKT ORDNAT.
- Ett binärt sökträd är SORTERAT och HIERARKISKT ORDNAT.
- En lista är linjärt ordnad (elementen följer en före-efter-relation) och är OSORTERAD (elementens position bestäms INTE av elementVÄRDENA).
- En stack är linjärt ordnat (elementen följer en före-efter-relation — eller över-under-relation om vi så vill) och är OSORTERAD.
- Ett heltal är en enkel datatyp så den är varken ordnad eller oordnad, varken sorterad eller osorterad.

Uppgift 8

Uppgift 8.1 Komplexitet för algoritm

Betrakta algoritmen för linjärsökning till vänster.

Låt varje funktionsanrop ha kostnader enligt tabellen nedan. Övriga operationer är gratis.

Operation	Kostnad
First()	1
Isend()	2
Value-isequal()	1
Inspect()	3
Next()	1
return från Linsearch	2

```
Algorithm Linsearch(l: List, v: Value, Value-isequal: Function)
// Input: An unsorted list, a search value, and a equality
//         function. The Value-isequal function should accept
//         two element values and return True if the values
//         are considered equal.
// Output: (True, pos), where pos is the position for the first
//         match, or (False, None) if no match is found.
p <- First(l)
while not Isend(p, l) do
    if Value-isequal(v, Inspect(p, l)) then
        return (True, p)
    p <- Next(p, l)
return (False, None)
```

Låt $T(n)$ vara funktionen som beskriver **värstafallskomplexiteten** för algoritmen som en funktion av antalet element n i listan.

	Cost	Count	
p <- First	1	1	1
Isend(p,l)	2	$n + 1$	$2n + 2$
Value-isequal	1	n	n
Inspect	3	n	$3n$
Next	1	n	n
return	2	1	2
Total			$7n + 5$

- Vilken av följande former får funktionen? ($an + b$)
- Ange värdena på koefficienterna a (7) och b (5)
- Vilken sorts komplexitet beskriver ovanstående fråga?
 - Absolut komplexitet, då vi tar hänsyn till tiden för alla operationer

Uppgift 8.2 Ordo

I uppgiften nedan, om något värde behöver avrundas, avrunda till heltal på lämpligt sätt. Om flera svar är möjliga krävs optimalt värde för fulla poäng.

Antag att en algoritm har komplexiteten $T(n) = 6n + 4$.

a) Är algoritmen $\mathcal{O}(n)$ enligt ordo-definitionen?

- Ja, med $c = 7$, $n_0 = 4$.

b) Är algoritmen $\mathcal{O}(n^2)$ enligt ordo-definitionen?

- Ja, med $c = 1$, $n_0 = 7$.

Uppgift 6 Pseudokod — Kopiering Fält av Riktad Lista

```
abstract datatype Array(val, index)
  Create(lo, hi: index) → Array(val, index)
  Low(a: Array(val, index)) → index
  High(a: Array(val, index)) → index
  Set-value(i: index, v: val, a: Array(val, index))
    → Array(val, index)
  Has-value(i: index, a: Array(val, index)) → Bool
  Inspect-value(i: index, a: Array(val, index)) → val

abstract datatype DList(val)
auxiliary pos
  Empty() → DList(val)
  Iseempty(l: DList(val)) → Bool
  First(l: DList(val)) → pos
  Next(p: pos, l: DList(val)) → pos
  Isend(p: pos, l: DList(val)) → Bool
  Inspect(p: pos, l: DList(val)) → val
  Insert(v: val, p: pos, l: DList(val))
    → (DList(val), pos)
  Remove(p: pos, l: DList(val)) → (DList(val), pos)
```

Skriv pseudokod för en funktion som skapar och returnerar en kopia av ett objekt av typen Fält av Riktad lista. Du får bara använda funktioner i gränsytan till datatyperna. Du ser gränsytorna till vänster. Du får anta att alla positioner i fältet innehåller en lista med noll eller flera element. Kopiering ska innehålla alla elementvärden i samma ordning.

Algoritmen ska ha följande huvud:

Algorithm CopyArrayofDList(a: Array)

Det är möjligt att *modularisera* lösningen, dvs. att identifiera en eller flera delalgoritmer/funktioner och beskriva dem separat. Ge dom i så fall lämpliga namn och kommentarer. Speciellt viktigt är kommentarer som beskriver inparametrar och returvärden. Tänk på att funktioner i pseudokod kan returnera flera parametrar.

Det är inte ett krav att modularisera lösningen, men det krävs för att ge fulla poäng.

Det som kommer att bedömas är

- att pseudokoden löser uppgiften under de givna förutsättningarna,
- att pseudokoden är fri från språkspecifika konstruktioner (inga `i++` eller `for i in range(...)`, etc.),
- att koden är korrekt indenterad,
- att koden är rimligt kommenterad,
- om koden är modulariserad, och
- om koden har optimal komplexitet ($g(n)$ är viktigast).

```

Algorithm CopyDList(l: DList)
// Return a copy of the directed list l

// Create an empty list
m ← Empty()

// Where to read in input list
p ← First(l)
// Where to insert in output list
// Will always be at the end
q ← First(m)
// For each element of the input list
while not Isend(p, l) do
    // Read value
    v ← Inspect(p, l)
    // Insert value at the end of the output list
    (m, q) ← Insert(q, v, m)
    // Advance position to be last in output list
    q ← Next(q, m)
    // Advance in input list
    p ← Next(p, l)

return m

Algorithm CopyArrayOfDList(a: Array)
// Return a copy of the array a. Each element of the array
// contains a directed list.

// Create an array with the same limits as a
c ← Create(Low(a), High(a))

// For each index in a
for i from Low(a) to High(a) do
    // Extract the list at i
    l ← Inspect-value(a, i)
    // Create a list copy
    m ← CopyDList(l)
    // Set at the proper position in the output array
    c ← Set-value(c, i, m)

// Return the array copy
return c

```

Bedömning

- Allvarliga fel (0p)
 - Sammanblandning av fält och lista => 0p
 - Använder index i lista => 0p
 - Blandar ihop index och positioner => 0p
 - Löser inte uppgiften => 0p
- Mellanallvarliga fel
 - Saknar funktion som kopierar lista (-6)
 - Python-specifika kommandon import (-6)
 - Stoppas in listan på index i en lista, ej i ett fält (-6)
 - Kopierar ej listorna (-6)
 - Sammanblandning av typer och variabler/funktionsnamn (variabelt avdrag)
 - pos och pos2 används i CreateCopy utan att skickas som parametrar (-4)
 - CopyDList förstör in-listan (odokumenterat) (-4)
 - Använder v,i på rad 3 utan att de är definierade (-4)
 - Ologiska for-loopar, saknar startvärde (-4)
 - Felaktig avläsning ([]) (-4)
 - Använder en position i två listor (-4)
 - Parametrar saknas på rad 2 (-4)
 - Minimalt med kommentarer (-3)
 - Ej modulär (-3)
 - CopyDList returnerar ibland en lista, ibland NULL (-3)
 - Returvärden sparas ej undan från t.ex. Next, Set-value, etc. (variabelt)