



TENTAMEN

Luleå tekniska universitet

Kurskod: D0009E
Kursnamn: Introduktion till programmering
Tentamensdatum: 151030
Skrivtid: 4 timmar
Tillåtna hjälpmedel: Inga

Jourhavande lärare m fullständigt telefonnr: Luleå: Fredrik Bengtsson, 0920492431	Jourhavande lärare m fullständigt telefonnr:
Jourhavande lärare m fullständigt telefonnr: Skellefteå: Patrik Holmlund, 0910585392	Jourhavande lärare m fullständigt telefonnr:

Betygsgränser:	
Totalt antal uppgifter och poäng:	7 uppgifter om totalt 37 poäng
Övriga upplysningar:	

Om du skriver tentan för första gången: Fyll i och lämna in kursvärdering i en separat hög!

Allmänna anvisningar

Kontrollera att du fått samtliga uppgifter.
Besvara endast en uppgift per Lösningsblad.
Skriv tydligt.

Efter tentamen

Tentamensresultatet syns "Mina sidor" på Studentwebben.
Resultat meddelas före sista anmälningssdag till nästa tentamenstillfälle.

Uppgifter till tryckeriet för tentor campus Luleå

Projektnummer SRT: 341980	Hur många ex: 140 (L) / 50 (S)
Hur många sidor: 4 inkl. försättsblad	Dubbel eller enkelsidigt: dubbel

Uppgift 1: (8p)

a:

Kommer följande program att terminera (terminera=avsluta)? Endast ja/nej-svar krävs.

```
i=0
while i>=0:
    if i<1:
        continue
    i = i+1
```

b: Vilken utskrift ger följande program:

```
a=[["a", "b"], ["c"]]
b="c"
c=2
if len(a)==c:
    print "Olle"
elif len(a+[[b]])<c and c>0:
    print "Maria"
else:
    print "nej"
```

c: Vilken utskrift ger följande program:

```
s=range(3,6)
st = "-"
for c in s:
    st = str(c) + st
print "-" + st
```

d:

Vad händer när följande kod exekveras (Om det blir fel, varför blir det fel?
Om det blir rätt, vilket värde får s?):

```
s=float(int("4"+"2")+0.2)+int("3")
s=int(s)+float("4")
```

Uppgift 2: (4 p)

Någon har skrivit en funktion, `cumulsum(ls)`, som är tänkt att ta en lista som parameter och uppdatera listan på följande sätt: elementet på position `n` i den uppdaterade listan ska innehålla summan av alla element från position 0 till och med `n` från den ursprungliga listan. M.a.o. varje tal i listan ska vara summan av alla talen från början fram till det talet.

Programmeraren kom fram till följande:

```
def cumulsum(ls):
    i=0
    while i<=len(ls):
        ls[i] = ls[i] + ls[i-1]
    return i
```

Programmeraren har gjort fel. Uppgiften är att rätta felen. Det är inte tillåtet att skriva ett helt nytt program, utan felen i detta befintliga måste rättas.

Uppgift 3: (5p)

Betrakta följande program:

```
def updateList(lst, value):
    if len(lst)>value:
        lst = lst + [value]
    return lst

list1 = [1,7,3]
list2 = updateList(list1, list1[0])
list2[0] = 8
list3 = updateList(list2, list2[0])
list2[0] = 2

print 'list1=',list1
print 'list2=',list2
print 'list3=',list3
```

Vilka värden får list1, list2 och list3 efter exekveringen?

Svara på formen:

```
list1 = ...
list2 = ...
list3 = ...
```

där punkterna är ersatta av svaren.

Uppgift 4: (5p)

Skriv en funktion `filterSpaces`, som tar en sträng som argument och returnerar en ny sträng där alla *whitespace* utom de inledande från argumentsträngen bytts ut mot understreck ("_"). Inledande betyder att ett tecken står före andra tecken och *whitespace* betyder att tecknet är ett tecken som flyttar markören utan att skriva nåt. En sträng med alla tecken som räknas som *whitespace* är `string.whitespace`.

Körexempel:

```
>>> filterSpaces("\n vi\t testar \rlite ")
'\n vi__testar__lite_'
```

Inbygda funktioner eller metoder för strängoperationer får inte användas. Det är tillåtet att använda funktionerna `len` och `range` samt operatoren `in`. Iteration kan ske på valfritt sätt med `for`, `while` eller rekursion.

Uppgift 5: (5p)

Betrakta följande kodsnuitt för att läsa in en lista av heltal från en fil till en lista:

```
def readList_():
    rList = []
    f = open("fil.txt", "r")
    while True:
        line = f.readLine()
        if line=="":
            return rList
        number = int(line)
        rList = rList + [number]
```

Modifiera funktionen enligt följande:

- Funktionen ska ta filnamnet på filen som ska öppnas som parameter.
- Om filen som ska öppnas inte existerar så ska tomma listan returneras.
- Om en rad i filen inte innehåller ett giltigt heltal så ska den raden hoppas över och inläsningen fortsätta med nästa rad.

Uppgift 6: (5 p)

Vi vill göra en klass `Statistics` som är tänkt att användas för att hålla koll på enkla statistiska mått hos numeriska värden. Klassen ska deklarera en metod `getMean` som ska returnera medelvärde av de hittills inmatade talen. En metod `getSum` ska returnera summan av de hittills inmatade talen. Slutligen ska en metod `addValue` finnas som tar som parameter ett nytt värde att behandla. Klassen ska också deklarera en initierare som inte tar några parametrar.
Tips: Det finns en mycket enkel intern representation som är lämplig för dessa metoder.

Uppgift 7: (5p)

Skriv en rekursiv funktion, `diffList`, som tar två listor av tal som parametrar och returnerar en lista som, för varje position `n`, i listorna, returnerar differensen mellan talet på position `n` i första listan och talet på position `n` i andra listan. Båda listorna kan antas vara av samma längd. Två tomma listor ska resultera i en tom lista.

Exempel:

```
>>> print diffList([6,3,2,9], [1,2,10,7])
[5, 1, -8, 2]
```