



TENTAMEN

Luleå tekniska universitet

Kurskod: D0009E
Kursnamn: Introduktion till programmering
Tentamensdatum: 171028
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:	

Fyll i kursvärderingen (frivilligt, men väldigt uppskattat!) och lämna in i 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 anmälningdag till nästa tentamenstillfälle.

Uppgifter till tryckeriet för tentor campus Luleå

Projektnummer SRT: 341980	Hur många ex: 130
Hur många sidor: 4 (tenta) + 7 (kursvärdering) = 8 totalt	Dubbel eller enkelsidigt: dubbel

Uppgift 1: (8p)

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

```
ls=[5,1,9,6,4,5]
while len(ls)==6:
    ls[0] = ls[0] - 1
```

- b:** Vilken utskrift ger följande program:

```
ls=[3,4,5]
b=7
c=6
if b>len(ls) or c>b:
    print "Första!"
elif c<b:
    print "Andra!"
else:
    print "Tredje!"
```

- c:** Vilken utskrift ger följande program:

```
s="Prog"
sn=""
for c in s:
    sn = c + sn
    print sn,
```

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

```
s="3"+str(int("345")+int(float("2.35")))
```

Uppgift 2: (4 p)

Någon har skrivit en funktion, `sumNegative(ls)`, som tar en lista som argument och returerar summan av alla negativa tal från listan. Programmeraren kom fram till följande:

```
def sumNegative(ls):
    i=0
    while i<=len(ls):
        if ls[i]<0:
            sum += ls[i]
            return sum
        i = i + 1
    print sum
```

Programmeraren har gjort fel. Uppgiften är att identifiera och 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 updateAndSelect(lst1, lst2, e):  
    if lst1[0]<lst2[0]:  
        lst1[0] = e  
        return lst2  
    else:  
        lst2[0] = e  
        return lst1  
  
lst1 = [6,5,4]  
lst2 = [3,2,1]  
lst2 = updateAndSelect(lst1, lst2, 7)  
lst3 = updateAndSelect(lst2, lst1, 0)  
lst1[2] = 8
```

Vilka värden får `lst1`, `lst2` och `lst3` efter exekveringen? Ange också vilka listor som är alias för varandra.

Svara på formen:

```
lst1 = ...  
lst2 = ...  
lst3 = ...
```

där punkterna är ersatta av svaren, samt skriv `lstX=lstY` om `lstX` och `lstY` är alias (X och Y siffror).

Uppgift 4: (5p)

Skriv en funktion `deleteFrom`, som tar två strängar, `st` och `delete`, som argument och returnerar en ny sträng där alla förekomster av alla tecken från `delete` är raderade från `st`.

Körexempel:

```
>>> deleteFrom("It's a secret to everybody.", "s be")  
"It'acrttovryody."
```

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

Uppgift 5: (5p)

Givet är följande funktion som läser in heltal från varje rad i en fil, adderar i till talet och skriver en ny fil med de nya talen.

```
def parseFile(i):
    fi = open("in.txt", "r")
    fo = open("ut.txt", "w")
    while True:
        row = fi.readline()
        if row=="":
            break
        fo.write(str(int(row)+i)+"\n")
    fi.close()
    fo.close()
```

Givet är också följande användargränssnitt till funktionen:

```
def parseUI():
    v = input("Enter number to add: ")
    parseFile(v)
```

a,

- Modifiera `parseFile` så att den kastar/ger undantaget `ReadError` om filen som läses från ("in.txt") inte existerar. Undantaget `ReadError` kan antas existera.
- Modifiera också funktionen så att rader som inte innehåller ett giltigt heltal hoppas över (inte skrivs till den nya filen). Undantagssystemet ska användas. Undantaget `ReadError` kan antas existera.
- Modifiera även så att filnamnen för filen som läses från och skrivs till resp. tas som parametrar till `parseFile`.

b, Modifiera `parseUI` så att "Processing failed." skrivs ut om filen som läses från ("in.txt") ej existerar. Filnamnen "in.txt" resp. "ut.txt" ska fortfarande användas. Undantagssystemet ska användas och den nya, modifierade `parseFile` ska användas för all filhantering. Det är tillåtet att använda `parseFile` som om uppgift (a) är löst, även om du inte löst uppgift (a).

Uppgift 6: (5 p)

Vi vill göra en klass `Vect2` för att representera tvådimensionella vektorer. En tvådimensionell vektor består av två tal, x och y . Klassen ska ha en konstruerare/initierare som tar x och y som parametrar. Klassen ska också deklarera en funktion som returnerar skalärprodukten av två vektorer. Skalärprodukten av två vektorer (x,y) och (a,b) är $x*a + y*b$. Dessutom ska två funktioner `getX` samt `getY` deklareras, som returnerar resp värde (x resp. y).

Uppgift 7: (5p)

Skriv en rekursiv funktion, `filterPos`, som tar en lista av tal, `lst`, som argument och returnerar en ny lista med endast de strikt positiva talen (ej 0). Exempel:

```
>>> filterPos([1,-1,-3,8,5,-2])
[1, 8, 5]
```