

i Information

Tentamen DVA245

Inga hjälpmedel är tillåtna. Länk till python-dokumentationen finns nere till vänster.

Max antal poäng: 26

För betyg 3 krävs 13p

För betyg 4 krävs 19,5p

För betyg 5 krävs 23,5p

Du kan få delpoäng på uppgifterna även om allt inte är helt rätt. Lycka till!

1 Pseudokod, tid, summa av kvadrater

a) Skriv i python funktionen som beskrivs av pseudokoden nedan. Pseudokoden finns också i PDFen. (3p):

b) Vad är tidskomplexiteten för funktionen? Hur beror tiden på antalet element i `in_list`? Skriv som kommentarer i koden. (3p)

```
function: SUM_SQUARES(in_list) returns  
the sum of the squares of all elements in  
in_list
```

```
sum_squares ← 0  
for elem in in_list do  
    sum_squares ← sum_squares + elem2  
return sum_squares
```

Skriv in ditt svar här

| | |
|---|--|
| 1 | |
|---|--|

2 Funktionsanrop

Koden nedan och i PDFen är en funktion som vi arbetat med:

```
def fun(item, sorted_list):
    lo = 0
    hi = len(sorted_list) - 1
    while lo <= hi:
        mid = (lo+hi)//2
        if sorted_list[mid] == item:
            return True
        elif sorted_list[mid] < item:
            lo = mid + 1
        else:
            hi = mid - 1
    return False
```

a) Vi anropar den såhär: `fun(12, [1, 3, 5, 7, 9, 12, 17, 20])`.

Vad är `lo` och `hi` i varje varv i `while`-loopen?

Vad returnerar anropet? (3p)

b) Vi anropar den såhär: `fun(2, [1, 3, 5, 7, 9, 12, 17, 20])`.

Vad är `lo` och `hi` i varje varv i `while`-loopen?

Vad returnerar anropet? (3p)

c) Vad är tidskomplexiteten för `fun`? (Hur beror tiden för `fun` på antalet element i `sorted_list`?) (3p)

d) Vilken funktion är det? (1p)

Skriv in ditt svar här

Teckenf...
B I U x₂ x² I_x
📄 📁 ↶ ↷ ↺
1= :≡ ≡≡ ≡≡≡
≡≡≡ ≡≡≡ Ω 📊 ✎ Σ ✖

Ord: 0

Totalpoäng: 10

3 Prioritetskö

Vi har en prioritetskö-klass med operationerna:

insert(priority, element) och

remove() som tar bort och returnerar ett element från kön.

Förklara hur du kan använda den för att sortera listan [12, 3, 5, 9, 4, 7].

(4p)

Skriv in ditt svar här

Teckenf... ▾ | **B** *I* U \times_2 \times^2 | \mathcal{I}_x | | | | | | | Σ |

Ord: 0

Totalpoäng: 4

Ord: 0

Totalpoäng: 6

Question 1

Attached



function: SUM_SQUARES(*in_list*) **returns**
the sum of the squares of all elements in
in_list

```
sum_squares ← 0  
for elem in in_list do  
    sum_squares ← sum_squares + elem2  
return sum_squares
```

Question 2

Attached



```
def fun(item, sorted_list):  
    lo = 0  
    hi = len(sorted_list) - 1  
    while lo <= hi:  
        mid = (lo+hi)//2  
        if sorted_list[mid] == item:  
            return True  
        elif sorted_list[mid] < item:  
            lo = mid + 1  
        else:  
            hi = mid - 1  
    return False
```