

Seminarium 01

Experimentering

7 september 2021

Lite om mig

Gustav Sörnäs

`gusso230@student.liu.se`

Ett år på Y, numera D3.

Handledare för halva U1.b.

Lite om er

D1 och U1 (och kanske några äldre).

Inga förväntade förkunskaper innan universitetet.

Redo att testa er fram själva / i små grupper med egen dator.

Lite om seminarieformen

Mer fokus på eget kodskrivande än "Ett steg i taget", men mer konkret än "Diskussion och analys".

Fritt fram att byta spår under terminen.

Innan seminariet: läs förberedelsematerialet och försök er på uppgifterna.

Lite om seminarieformen – skicka in kod

Några uppgifter innebär att skriva kod. (Experimentera!)

För att skicka in kod har vi två alternativ.

1. Egenutvecklad grej jag gjorde. Inte testat än. Klistra in er kod, skriv ert namn och tryck på "Submit".
2. Google Drive (backup): Om min grej inte fungerar (högst troligt) får ni klistra in kod i ett gemensamt Google Drive-dokument. Vi löser länkproblemet om det kommer.

Dagens seminarium

- ▶ Satser och uttryck.
- ▶ Analysera `fibonacci`.
- ▶ Implementera en matematiskt definerad funktion.
- ▶ Uppgift: skriv en egen funktion.
- ▶ Variablers synlighet.

Inte så mycket kodskrivning. Diskussioner först i mindre grupper och sedan i helklass.

Satser och uttryck

Sats (*statement*): En typ av *instruktion* som kan utföras av Python. Enkla satser (return, tilldelning, ...) och sammansatta satser (for, def, if, ...).

Uttryck (*expression*): Något som Python kan utvärdera till ett värde. Enkla uttryck (identifierare och *literals* (eller *konstanter*)) och sammansatta uttryck (additioner, subtraktioner, ...).

fibonacci

```
1  def fibonacci(n):
2      if n < 3:
3          return n - 1
4      a = 0
5      b = 1
6
7      for i in range(n - 2):
8          c = b
9          b = a + b
10         a = c
11     return b
```

1. Hur många satser finns i koden? Var finns dom?
2. Hur många sammansatta uttryck finns i koden?

fibonacci

```
1  def fibonacci(n):  
2      if n < 3:  
3          return n - 1  
4      a = 0  
5      b = 1  
6  
7      for i in range(n - 2):  
8          c = b  
9          b = a + b  
10         a = c  
11     return b
```

1. Vad gör funktionen?
2. Vad returnerar funktionen?
3. Vilken indata hanterar funktionen?
4. Vad används variabeln c till? Vad skulle den kunna heta istället?
5. Är returvärdet ett heltal eller flyttal?
6. Vad kommer hända om ni matar in (olika) flyttal?

gcd

$$\text{gcd}(a, 0) = a$$
$$\text{gcd}(a, b) = \text{gcd}(b, a \bmod b)$$

1. Beskriv hur beräkningen går till.
2. Implementera gcd i Python.
3. Räkna antalet satser och sammansatta uttryck i er Python-kod.

seminarium.sörnäs.se

gcd

Förslag:

```
1 def gcd(a, b):  
2     if b == 0:  
3         return a  
4     return gcd(b, a % b)
```

1. Beskriv hur beräkningen går till.
2. Implementera gcd i Python.
3. Räkna antalet satser och sammansatta uttryck i er Python-kod.

seminarium.sörnäs.se

Uppgift: latest_birthday_before

I grupper om minst fem personer, skriv en funktion som tar in två heltal (månad och dag) och returnerar namnet på den i gruppen vars födelsedag är närmast i tiden, räknat *bakåt* från det inskickade datumet.

Uppgift: latest_birthday_before

I grupper om minst fem personer, skriv en funktion som tar in två heltal (månad och dag) och returnerar namnet på den i gruppen vars födelsedag är närmast i tiden, räknat *bakåt* från det inskickade datumet.

Exempel:

- | | |
|---------|--|
| 1. 20/8 | <code>latest_birthday_before(18, 8) -> (15, 8)</code> |
| 2. 15/8 | <code>latest_birthday_before(15, 8) -> (15, 8)</code> |
| 3. 18/7 | <code>latest_birthday_before(14, 8) -> (18, 7)</code> |

Uppgift: latest_birthday_before

I grupper om minst fem personer, skriv en funktion som tar in två heltal (månad och dag) och returnerar namnet på den i gruppen vars födelsedag är närmast i tiden, räknat *bakåt* från det inskickade datumet.

Exempel:

- | | |
|---------|--|
| 1. 20/8 | <code>latest_birthday_before(18, 8) -> (15, 8)</code> |
| 2. 15/8 | <code>latest_birthday_before(15, 8) -> (15, 8)</code> |
| | <code>latest_birthday_before(14, 8) -> (18, 7)</code> |
| 3. 18/7 | <code>latest_birthday_before(10, 7) -> ?</code> |

Uppgift:

```
1  g = 0
2  def outer(in1):
3      a = "a"
4      def inner(in2):
5          print(a) # 1
6          c = "c"
7          a = "a from inner"
8          print(in2) # 2
9      print(c) # 3
10     print(a) # 4
11     inner("into inner")
12     print(c) # 5
13     print(a) # 6
14     if g == 0:
15         d = "g = 0"
16     else:
17         e = "g != 0"
18     print(d) # 7
19     print(e) # 8
```

1. Vilka print-satser kommer fungera? Vilka kommer att krascha?
2. Vad skriver dom som fungerar ut?