

Tenta i Grundläggande programmering
DD1331 2016-10-20 klockan 14.00–16.00

Tillåtna hjälpmedel: En Pythonbok, skrivmaterial, mat, medicin och vattenflaska.

Otillåtna hjälpmedel: Laptop, mobiltelefon eller miniräknare.

Bonuspoäng (upp till 4 för labbar redovisade i tid) gäller. För godkänt krävs 14 poäng. Högsta poängen du kan få är 24 (20 tentapoäng och 4 bonuspoäng).

Hur du fyller i dina svar

Skriv dina svar på ett separat skrivpapper. Fyll i efternamn, förnamn och personnummer i de angivna fälten längst upp och svara sedan på 11 rader med uppgiftsnummer till vänster följt av det eller de svarsalternativ som du tror är rätt. Exempel:

- 1 f g
- 2 a c
- 3 o
- 4 a
- 5 f
- 6 b
- 7 c
- 8 e
- 9 f h
- 10 q
- 11 z

När du går, lämna in svarspapperet och välj själv om du vill behålla frågorna (för att stämma av dina svar med facit) eller lämna dem till pappersåtervinning. Tentan och facit kommer att publiceras på Kth Social. Lycka till!

MVH
Marcus

Tenta i DD1331 Grundläggande Programmering

1. (1p)

Följande uttryck evalueras i Idle. Vad blir resultatet?

```
>>> "1" + 2
```

- (a) Ettan konverteras till en int och svaret blir 3.
- (b) Tvåan konverteras till en sträng och svaret blir "12".
- (c) Det blir typfel.

2. (2p)

Följande uttryck evalueras i Idle. Vad blir resultatet?

```
>>> 1.0/2/2
```

- (a) 0.0
- (b) 0.25
- (c) 1.0
- (d) Inget av ovanstående alternativ.

3. (1p)

Följande Python3-program definierar en klass för en boll. Vad behövs för att skapa ett objekt av klassen?

```
class Boll:
    def __init__(self):
        self.x, self.y, self.z = 0,0,0

def main():
    # Ersätt denna kommentarsrad med koden för att skapa en Boll.

main()
```

- (a) minBoll = Boll()
- (b) minBoll = Boll("Anna")
- (c) minBoll = new Boll()
- (d) minBoll = new Boll("Anna")

4. (2p)

Vad svarar Idle efter att dessa rader Python3 har matats in?

```
>>> lista = [3,1,4,1,5,9,2,6]
>>> lista[1]+lista[-1]
```

- (a) 1
- (b) 2
- (c) 3
- (d) 4
- (e) 5
- (f) 6
- (g) 7
- (h) 8
- (i) 9
- (j) Inget av ovanstående alternativ.

5. (2p)

Vad skriver följande Python3-program ut?

```
def mystery(n):
    if n <= 0:
        return 0
    else:
        return n+mystery(n-2)

print(mystery(5))
```

- (a) 1
- (b) 2
- (c) 3
- (d) 4
- (e) 5
- (f) 6
- (g) 7
- (h) 8
- (i) 9
- (j) Inget av ovanstående alternativ.

6. (2p)

Vad skriver följande Python3-program ut?

```
def mystery(a,b):  
    a,b = b,a  
    print(a,b, end = " ")
```

```
def main():  
    a,b = 13,42  
    mystery(a,b)  
    print(a,b)
```

```
main()
```

- (a) 13 42 13 42
- (b) 13 13 13 13
- (c) 42 42 42 42
- (d) 13 42 42 13
- (e) 42 13 13 42
- (f) 42 13 42 13

7. (2p)

Vad skriver följande Python3-program ut?

```
def main():  
    fourcount = 4  
    while fourcount <= 20:  
        print(fourcount, end = " ")  
        fourcount = fourcount + 4  
    print()
```

```
main()
```

- (a) Ingenting.
- (b) 4 8 12 16
- (c) 4 8 12 16 20
- (d) 4 8 12 16 20 24
- (e) En oändlig lista med fyror och mellanslag.

8. (2p)

En sorterad lista innehåller 1000 heltal. Du vill undersöka om talet 997 finns med i listan och använder binärsökning. Hur många tal i listan kommer binärsökningen maximalt att läsa av?

- (a) 1 element
- (b) Ungefär 10 element
- (c) Ungefär 100 element
- (d) Alla 1000 elementen

9. (2p)

Hur många tal skriver följande Python3-program ut?

```
def main():  
    for i in range(5):  
        print(i)
```

```
main()
```

- (a) 1
- (b) 2
- (c) 3
- (d) 4
- (e) 5
- (f) 6
- (g) 7
- (h) Fler än 7.
- (i) Inga alls.

10. (2p)

Vad skriver följande Python3-program ut?

```
def mystery(n):  
    if n <= 0:  
        return 1  
    else:  
        return n*mystery(n-1)
```

```
print(mystery(3))
```

- (a) 1
- (b) 3
- (c) 6
- (d) 0
- (e) Ingenting alls. Det har ett syntaxfel.
- (f) Ingenting alls. Programmet fortsätter för evigt.

11. (2p)

Vad skriver följande Python3-program ut?

```
def mystery(n):  
    return mystery_helper(n,n)  
  
def mystery_helper(x,y):  
    if y == 0:  
        return 1  
    if y > x:  
        return 0  
    return mystery_helper(x,y-1) + mystery_helper(x-1,y)
```

```
print(mystery(3))
```

- (a) 1
- (b) 2
- (c) 3
- (d) 4
- (e) 5
- (f) 6
- (g) 7
- (h) 8
- (i) 9