

- Tentan har flervalsfrågor där minst ett svarsalternativ är korrekt. Om man svarar fel eller inte har exakt antal rätta alternativ får man noll poäng på frågan.
- Man måste bli godkänd på del A (4 rätt på 8 frågor) för att del B ska rättas.
- Del B består av frågor med varierande poäng (totalt 12 poäng).
- Inga import (Pythons standardbibliotek eller externa bibliotek) får användas om de inte nämns eller finns med i uppgiften. Man får använda inbyggda funktioner som `len`, `range` och `map`.
- All kod avser **Python 3**, dvs *inte* t.ex. Python 2.7
- **Hjälpmedel:** Ett A4 med så mycket information du vill. Du får skriva på båda sidorna.
- **Betygsgränser:** E: 10, D: 12, C: 14, B: 16, A: 18, av maximala 20.

Del A: flervalsfrågor

Var snäll samla svaren på del A på ett svarspapper.

1. Vad blir resultatet av koden till höger?

- A. 5
- B. 7
- C. 5, 7
- D. `5 == 7`
- E. False
- F. Ett felmeddelande

```
x = 7
y = 5
print(x == y)
```

2. Vad blir resultatet av koden till höger?

- A. DA2005
- B. MM2001
- C. 7.5
- D. 30
- E. Ett felmeddelande

```
dct = {'DA2005': 7.5, 'MM2001': 30}
print(dct[0])
```

3. Vad blir resultatet av koden till höger?

- A. 'H'
- B. 'HUBBA'
- C. ['H', 'U', 'B', 'B', 'A']
- D. [72, 117, 98, 98, 97]
- E. Ett felmeddelande.

```
[c.upper() for c in 'Hubba']
```

4. Vilken inbyggd Python-funktion är mest lik `fcn` som definieras här?

- A. `enumerate`
- B. `filter`
- C. `map`
- D. `reduce`
- E. `zip`

```
def fcn(f, a):
    res = []
    for elem in a:
        if f(elem):
            res.append(elem)
    return res
```

5. Vilka särfall *kan* uppstå när man använder funktionen huh?

- A. KeyError
- B. SyntaxError
- C. TypeError
- D. ValueError
- E. ZeroDivisionError

```
def huh(dct):  
    s = 0  
    for x in dct:  
        y = dct[x]  
        s += float(x) / float(y)  
    return s
```

6. Betrakta koden till höger. Hur många gånger skrivs ordet 'ping' ut?

- A. 1
- B. 2
- C. 3
- D. 5
- E. Ingen gång, det blir ett fel.

```
def f(x):  
    if x > 0:  
        print('ping')  
        g(x-1)  
    else:  
        print('Jag vann')  
def g(x):  
    if x > 0:  
        print('pong')  
        f(x-1)  
    else:  
        print('Du förlorade')  
f(5)
```

7. Vad blir resultatet av koden till höger?

- A. 0.0
- B. 0.5
- C. 1.0
- D. False
- E. Ett felmeddelande

```
def fcn(s1, s2):  
    x = 0  
    n = 0  
    for i in range(len(s1)):  
        if s1[i] == s2[i]:  
            x += 1  
            n += 1  
    return x / n  
print(fcn('DA2005', 'MM2001'))
```

8. Vad är resultatet av koden till höger?

- A. -17
- B. -3
- C. -2
- D. 0
- E. Värdet None
- F. Inget, det blir ett felmeddelande.

```
def max(lst):  
    val = 0  
    for elem in lst:  
        if elem > val:  
            val = elem  
    return val  
print(max([-3, -2, -17]))
```

Del B: kodfrågor

Var snäll använd ett papper till varje fråga i del B. Delfrågor, som 10A och 10B, får gärna lösas på samma papper.

9.

- A. Skriv funktionen `transmogrifier(dct)` som tar en uppslagstabell `dct` och returnerar en ny uppslagstabell där nycklar och värden har bytt plats jämfört med `dct`. Funktionen ska klara av ”enkla” uppslagstabeller, likt de som ges i exemplet nedan. (2p)
- B. Ge ett exempel på en uppslagstabell som `transmogrifier(dct)` inte kan förväntas klara av, i den mening att det ”måste” bli ett körfel (”run-time error”). Motivera ditt svar. (1p)

Exempelanvändning:

```
[In: ] transmogrifier({})
[Out:] {}
[In: ] transmogrifier({'DA2004': 7.5})
[Out:] {7.5: 'DA2004'}
[In: ] transmogrifier({'a': 1, 'c': 73, 'x': 20})
[Out:] {1: 'a', 73: 'c', 20: 'x'}
```

10. Definiera funktionen `incrementer(lst)` som, givet att `lst` är en lista (eller annan itererbar datastruktur) med tal, returnerar samma serie tal men med 1 adderat. **Krav:** Din lösning ska använda `map` tillsammans med en *anonym funktion* för att addera talet 1. (2p)

Exempelanvändning:

```
[In: ] list(incrementer([]))
[Out:] []
[In: ] list(incrementer([1,2,3]))
[Out:] [2, 3, 4]
[In: ] list(incrementer([-3, 17.5, 0, -2.25]))
[Out:] [-2, 18.5, 1, -1.25]
```

11.

- A. Skriv funktionen `is_prime(n)` som returnerar `True` om n är ett primtal och annars `False`. Algoritmen du ska använda ska prova att dela n med varje tal mellan 2 och $\lfloor \sqrt{n} \rfloor$. Om funktionen hittar en delare så ska `False` genast returneras, men om inget av heltalen från 2 till $\lfloor \sqrt{n} \rfloor$ är en delare till n så ska `True` returneras. För $n < 2$ ska funktionen genast returnera `False`.
Kom ihåg att $\lfloor x \rfloor$ betyder ”avrundat nedåt”, så $\lfloor 2.9 \rfloor = 2$. Du ska använda funktionerna `floor` och `sqrt` från modulen `math` som avrundar nedåt respektive beräknar kvadratroten. (2p)

Exempelanvändning:

```
[In: ] is_prime(1)
[Out:] False
[In: ] is_prime(2)
[Out:] True
[In: ] is_prime(3)
[Out:] True
[In: ] is_prime(4)
[Out:] False
[In: ] is_prime(17)
[Out:] True
```

- B. Lägg funktionalitet till så att `is_prime(n)` ger ett `ValueError` om talet n inte är ett heltal. (1p)

12. Skriv generatorn `all_primes()` som, givet tillräckligt med tid och minne, genererar alla primtal. Det är antagligen en bra idé att använda `is_prime` från fråga 11 som hjälpfunktion, och det kan du göra utan att ha löst den uppgiften. (2p)

Din lösning ska fungera som i exemplet:

```
[In: ] for p in all_primes():
        print(p, end=' ')
        if p > 30:
            break
        print()
[Out:] 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31
```

13. Skriv klassen `Color` för att representera färger givna i färgsystemet RGB. Klassens konstruktor ska ta värden mellan 0 och 1 för färgerna röd, grön och blå. Det ska finnas en metod `intensity()` som returnerar genomsnittet av de tre färgkomponenterna. (2p)

Exempelanvändning: Programmet

```
black = Color(0, 0, 0)
white = Color(1, 1, 1)
red = Color(1, 0, 0)
su_blue = Color(0, 0.18, 0.37) # Universitets profilmfärg
print('Black intensity:', black.intensity())
print('White intensity:', white.intensity())
print('Red intensity:', red.intensity())
print('Blue intensity:', su_blue.intensity())
```

ska ge utdata:

```
Black intensity: 0.0
White intensity: 1.0
Red intensity: 0.3333333333333333
Blue intensity: 0.18333333333333335
```