

- **Skriv tydligt.** Svårlästa svar riskerar 0 poäng.
- Skriv bara på en sida av varje papper!
- 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 (5 rätt på 10 frågor) för att del B ska rättas.
- Inga externa bibliotek får användas om de inte nämns eller finns med i uppgiften, man får dock använda inbyggda funktioner som `len` och `range`.
- **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. Vilka utav följande påståenden är korrekta baserat på information från kodraden till höger?

- X är en klass.
- A är en klass.
- X är en basklass(/superklass) till A.
- A är en basklass(/superklass) till X.
- X ärver metoder från A.
- A ärver metoder från X.

```
class X(A):
```

2. I vilken ordning skrivs tuplerna ut av koden till höger?

- (1,a), (1,b), (2,a), (2,b)
- (1,b), (1,a), (2,b), (2,a)
- (2,a), (2,b), (1,a), (1,b)
- (2,b), (2,a), (1,b), (1,a)

```
for x in [1,2]:  
    for y in ['a','b']:  
        print((x,y))
```

3. Vilket resultat skrivs ut av koden till höger?

- 10
- 15
- 20
- 30
- Ingenting, det blir ett särfall.

```
def g(f,a,b):  
    for i in range(a,b+1):  
        yield f(i)  
  
print(sum(g(lambda x: 2*x, 1,5)))
```

4. Vad blir det för utskrift av koden till höger?

- 12
- 1
- 3
- 8
- 15
- Ingenting, det blir ett särfall.

```
x=1  
def f(x, y):  
    y = 5  
    return x + y  
  
x=10  
y=-15  
print(f(3,0))
```

5. Vad blir det för utskrift av koden till höger?

- a. 3.
- b. 4.
- c. 5.
- d. 6.

```
l = [ [], [1], [[1], [2], [3,4]] ]  
print(sum([len(item) for item in l]))
```

e. Ett särfall.

6. Vad är syftet med `continue`?

- a. Att omedelbart gå till början av en for/while loop.
- b. Att stänga fil för läsning/skrivning.
- c. Att avsluta en loop i förtid.
- d. Att orsaka radbrytning efter en print-sats.
- e. Inget, instruktionen finns inte i Python.

7. Vilket eller vilka påståenden är sanna?

- a. Funktioner returnerar `None` om return inte angivits i funktionskroppen.
- b. Funktioner kan returnera flera värden i en `return`-sats.
- c. En dokumentationssträng måste ligga omedelbart efter funktionshuvudet (dvs efter `def` funktion) om den ska ha tänkt funktionalitet.
- d. Funktioner kan kalla på funktioner.

8. Vad är resultatet av satsen `print([3*x for y in range(3)])`?

- a. [0, 1, 2]
- b. [0, 3, 6]
- c. [0, [0,1,2], [0,1,2]]
- d. [[0,1,2], [0, 3, 6], [0, 3, 9]]
- e. Ett särfall

9. Vad blir det för utskrift av koden till höger?

- a. [(0, 3), (1, 2), (2, 1)]
- b. [(0, 3), (1, 2)]
- c. [(0, 3)]
- d. [(3, 0), (2, 1), (1, 2)]
- e. [(3, 0), (2, 1)]
- f. [(3, 0)]

```
i, j, l = 3, 0, []  
while j < i:  
    l.append((i,j))  
    i -= 1  
    j += 1  
print(l)
```

10. Vilken eller vilka boolska variabeltilldelningar gör att uttrycket

```
((not x) and (y or z)) or (x and (y or z))
```

evaluerar till `True`?

- a. x=False, y=False, z=False
- b. x=True, y=False, z=False
- c. x=False, y=True, z=False
- d. x=True, y=False, z=True
- e. x=False, y=False, z=True

## Del B: kodfrågor

11. För att komma åt värdet 12 i nedan kod skriver man `x[0]`. Hur ska man kalla på `x` för att komma åt värdet `secret` i koden nedan?

```
d = {"password": "secret", "hej": 2 }
x=[12, 93, ["a", [d]]]
```

12. I koden till höger finns en funktion som genererar en sekvens av Collatz-tal givet ett tal `n`. Skriv om koden så att (1) funktionen tar en parameter `max_iter`, och (2) returnerar efter högst `max_iter` iterationer. Med hjälp av den extra parametern ska man alltså kunna begränsa antalet iterationer i händelse av lång sekvens.

```
def collatz(n):
    while n != 1:
        yield n
        if n % 2 == 0:
            n = n/2
        else:
            n = 3*n + 1
```

13. Skriv en funktion `indices` som returnerar en lista med index på alla samma element i en lista. Till exempel, om `l = [True, False, True, True, False]` ska `indices` returnera `l = [0, 2, 3]`

14. Skriv en rekursiv funktion `strip_rec(s)` som tar bort alla mellanslag i slutet på `s` och returnerar denna modifierade sträng. Till exempel, om `s = 'abc '` så ska `strip_rec(s)` returnera `'abc'`.

15. Betrakta koden till höger. Det kan uppstå ett eller flera särfall (eng. exception) vid anrop till `g(x)`. Skriv om koden så att `g(x)` returnerar `None` istället för att särfall uppstår. `math.fabs(x)` returnerar absolutbeloppet (flyttal) av talet `x` och `math.sqrt(x)` returnerar kvadratroten (flyttal) av `x`.

```
import math
def g(x):
    if math.fabs(x) < 1:
        return 1/(x + 0.5)
    else:
        return math.sqrt(x)
```

16. Skriv en funktion `count_spaces(f)` som tar in ett filnamn `f` och räknar mellanslag.

Om vi har en textfil som heter `fil.txt` med innehåll som i rutan till höger ska funktionen fungera på följande sätt:

```
>>> count_spaces('fil.txt')
4
```

```
hej hopp
penna
papper
lycka till pa tentan
```

17. Antag en uppslagstabell `d` som inte nödvändigtvis innehåller unika värden associerat med nycklarna. Skriv en funktion `reformat_dict` som tar `d` och returnerar en ny uppslagstabell `d_new` där alla värden i `d` är unika nycklar i `d_new` och värdena i `d_new` är listor med alla nycklar som associerades men värdet i `d`. Exempel användning

```
d = {1 : "a", 2 : "a", 3 : "b", 4 : "a"}
reformat_dict(d) # returnerar {'a': [1, 2, 4], 'b': [3]}
```

18. Skriv en klass `Circle` för att representera cirklar i planet. Dess konstruktor ska ta en `x`- och en `y`-koordinat samt radien `r`.

19. Visa hur man instansierar två `Circle`-objekt till variablerna `c1` och `c2`.

20. Implementera metoden `__lt__` (d.v.s "less than") i klassen `Circle` som ska jämföra arean av två cirklar och returnera sant om `c1` har mindre area än `c2`. (Notera att `__lt__` är en utav pythons reserverade funktionsnamn som gör att denna metod kan anropas som `c1 < c2`)