

# Övningstenta - Programmering med Python

---

För G krävs 10 poäng och för VG krävs 15 poäng.

## Uppgift 1 (3p)

Beskriv vad koden gör och vad som skrivs ut:

```
1. frukter = ["jordgubbe", "vindruva", "grape"]
2. godfrukt = frukter[2]
3. print(godfrukt[-1])
```

Skriv en kort kommentar för varje rad.

## Uppgift 2 (3p)

Skriv en funktion som tar in en radie och en höjd som inparametrar och som returnerar volymen av en cylinder.

Volymen för en cylinder är:

$$V = \pi r^2 h$$

Låt användaren mata in en radie och en höjd och kör funktionen. Tänk på att ta med lämplig felhantering.

## Uppgift 3 (3p)

Simulera 1000000 tärningskast och beräkna antalet ettor, tvåor, treor, fyror, femmor och sexor. Spara resultatet i en lämplig datastruktur.

## Uppgift 4 (3p)

Skapa en funktion som tar in ett heltal n som parameter och skriver ut mönstret nedan.

Exempel när n = 5:

```
x x x x x
x o x o x
x x x x x
x o x o x
x x x x x
```

## Uppgift 5 (4p)

Skapa en funktion som tar in en summa pengar.

Den ska printa ut en representation av sedlar och mynt som representerar summan.

Exempel 3214 ska ge:

```
Tusenlapp: 3
Tvåhundra: 1
Tiokrona: 1
Enkrona: 4
```

## Uppgift 6 (4p)

Du har följande kod:

```
class Animal:
    def __init__(self, name):
        self.name = name

    def __str__(self):
        return f"Animal {self.name}"

class Dog(Animal):
    # notera att __init__ är nedärvd
    def __repr__(self):
        return f"Dog: {self.name}"

class Fish(Animal):
    def __repr__(self):
        return f"Fish: {self.name}"

# TO-DO: implementera PetOwner
class PetOwner:
    ...

# manuell test av PetOwner
owner1 = PetOwner("Ada", ["Snoopy", "Pluto"], ["Fishy", "Buppy"])
owner2 = PetOwner("Beda", ["Barkly"] )
print(owner1)
print(owner2)
```

Implementera PetOwner klassen så att du får liknande utskrift som nedan med det manuella testet.

```
Ada owns:  
[Dog: Snoopy, Dog: Pluto]  
[Fish: Fishy, Fish: Buppy]  
Beda owns:  
[Dog: Barkly]
```