

# Objektorientering 3

vt 25

# Innehåll

Klasser  
Repetition

Övningen  
Mål  
Klassdiagram  
Turtle

# Vad är en klass?

- ▶ *Klasser* i programmering är en sorts *abstraktion*
- ▶ Används för att *modellera* verkligheten

# Objektifiering

Om vi skulle vilja modellera en sköldpadda så behöver vi bestämma vilka egenskaper en sköldpadda kan ha:

- ▶ Attribut, så som färg, födelseår, namn m.m.
- ▶ Förmågor, så som gå, äta m.m.

# Innehåll

Klasser  
Repetition

Övningen  
Mål  
Klassdiagram  
Turtle

# Dagens mål

- ▶ Under dagens pass ska vi skapa en sköldpadda som kan gå över skärmen och styras för att äta upp godisbitar
- ▶ Vi kommer att behöva skapa två klasser för detta
  1. Sköldpadda
  2. Godis
- ▶ Båda klasserna har en del gemensamt:
  - ▶ Position på skärmen
  - ▶ Förmåga att ritas ut

# Klassen sköldpadda

- Klassen sköldpadda kommer att behöva innehålla följande:

Sköldpadda
x: int y: int hastighet: int riktning: int
gå(): upp(): ner(): vänster(): höger(): rita():

# Klassen godis

- Klassen godis kommer att behöva innehålla följande:

Godis
x: int y: int
ny_position(): rita():



## Turtle-specifikt

- För att rita ut på skärmen behöver våra två klasser några Turtle-specifika rader kod

```
1 # Följande ska ligga i __init__  
2 self.turtle = turtle.Turtle() # Skapar något att ha på skärmen  
3 self.turtle.shape('turtle') # alternativt 'square' i godis  
4 self.turtle.pu() # Lyfter pennan såp att vi inte ritar streck
```

```
1 def rita(self):  
2     self.turtle.seth(self.riktning*90) # Behövs inte i Godis  
3     self.turtle.setpos((self.x, self.y)) # Uppdatera positionen  
        på skärmen
```

## Turtle-specifikt

- Dessutom behöver vi ha lite mer kod för att programmet ska göra som vi vill

```
1 import turtle
2 import time
3 window = turtle.Screen() # Skapar fönstret
4 window.tracer(0) # Gör att saker händer direkt
5 window.listen() # Lyssnar efter knapptryck
6 window.onkey(sköldis.vänster, 'a') # Om man trycker på a...
7 ...
8 while True:
9     time.sleep(0.2) # Gör att allt inte går för snabbt
10    window.update() # Uppdaterar fönstret
11    sköldis.move() # Flyttar sköldpaddan
12    sköldis.draw() # Ritar sköldpaddan
13    candy.draw() # Ritar godiset
```

# Övningar

1. Implementera klassen Sköldpadda
2. Implementera klassen Godis
3. Skapa en funktion som kollar om sköldpaddan nuddar godiset
4. Flytta på godiset efter att sköldpaddan ätit upp det.
5. Lägg till att man inte kan vända sig om 180 grader.
6. Lägg till en ruta som följer efter den första rutan när man ätit en godis.