

PROGRAMMERINGSTEKNIK

FÖRELÄSNING 8

IDAG:

- Avlusning
- Dela upp i funktioner
- Defaultparametrar
- Egna moduler



OLIKA TYPER AV FEL



- Felavbrott (Exception) när programet körs
- Inget händer när man kör programmet
- Massor av text rinner över skärmen
- Programmet gör något annat än det man ville
- Programmet gör rätt för vissa indata, men inte för andra

FELAVBROTT

Lär dig tolka felutskriften!

Traceback (most recent call last):

```
File "filmer.py", line 124, in  
<module>
```

```
titta(listan)
```

```
File "filmer.py", line 102, in titta
```

```
film.ny_visning(1)
```

```
TypeError: ny_visning() takes exactly 1  
argument (2 given)
```

TOLKNING

- Sista raden förklarar felet!

```
TypeError: ny_visning() takes exactly 1  
argument (2 given)
```

- Raderna ovanför visar anropskedjan. Läs nerifrån och uppåt! Felet uppstod på rad 102, i funktionen titta som anropades på rad 124.

KONTROLLUTSKRIFTER

- Använd kontrollutskrifter för att hitta var i programmet felet uppstår.
- En kontrollutskrift är en vanlig print-sats, till exempel:

```
print("Klar med inläsningen")
```
- Eller stanna upp så här:

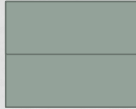
```
input("Tryck Enter")
```
- Du kan också skriva ut variabelvärden för att se hur dom ändras under körning.

```
input("x=", x, " (Enter)")
```

MITTIPRICK-METODEN

Anta att programmet hänger sig, men vi vet inte var i programmet det inträffar.

1. Lägg en kontrollutskrift i början och en i slutet. Blev det fel däremellan?
2. Lägg då in en kontrollutskrift mitt i. Om den kommer ut som den ska finns felet i andra halvan, annars i första.
3. Fortsätt tills du hittat felet!



DELA UPP I FUNKTIONER

- Hitta sätser som hör ihop
- Vad ska funktionen heta?
- Vilka variabler ska vara parametrar?
- Vilka variabler ska skickas ut som returvärde?

```
n=int(input("Hur många djur vill du ha?"))

#Skapar en lista med n stycken husdjur
lista = []
for i in range(n):
    nytt = Husdjur()
    lista.append(nytt)

#Visar alla husdjur
n = len(lista)
print("\n*** Du har", n, "husdjur: ***")
for djur in lista:
    print(djur)
print("*****\n")
```

FÖRSTA FUNKTIONEN

```
def skapaLista(n):
    #Skapar en lista med n stycken husdjur
    lista = []
    for i in range(n):
        nytt = Husdjur()
        lista.append(nytt)
    return lista
```

NYA HUVUDPROGRAMMET

```
n=int(input("Hur många djur vill du ha? "))

lista = skapaLista(n)

visa(lista)

slinga(lista)

adjö(lista)

Alla metoder använder lista. Vore det inte enklare att ha en
klass med lista som attribut?
```

NY KLASS: BUR

```
class Bur(object):
    # Flera virtuella husdjur i en bur

    def __init__(self, n):
        """Skapar en lista med n Husdjur"""
        self.lista = []
        for i in range(n):
            self.lista.append(Husdjur())
```

...FLER METODER I BUR

```
def banna(self):  
    for djur in self.lista:  
        djur.banna()  
  
def mata(self, bullar):  
    for djur in self.lista:  
        djur.mata(bullar)  
  
def leka(self):  
    for djur in self.lista:  
        djur.leka()
```