

# GruProg DD1331

## Övning 6

Anna Karlhede


karlhed@kth.se

Repo: <https://github.com/TypAnna/GruProgDD1331>

# Idag

- Repetition/genomgång
  - Tentan
- Uppgifter
  - funktioner & scope
  - minnesallokering
  - rekursion

# Repetition & genomgång



# Tentan

Underskatta inte tentan - lägg ner ordentlig med tid.

Om (när...) ni gör extentor - gör varje uppgift grundligt och kolla inte på facit direkt.

- extra viktigt eftersom det finns så få extentor (finns faktiskt ingen som har erat format)

Boken är tillåten på tentan.

Gå in [här](#) och läs N-O-G-A.

Gör uppgifter från boken och på [Kattis](#).

Leta upp uppgifter på nätet (programmering ät tacksamt på det viset)

Gå igenom powerpointsen (Dicanders och dessa, men kom ihåg att iaf dessa EJ innehåller allt i kursen, tex **inget** om von Neumann!)

Ni måste kunna tillräckligt mycket om ALLT för att klara tentan. Om du missar helt på ett lärandemål så kan du inte få godkänt.

# Svårare koncept som vi gått igenom

Mutable/immutable datatyper och vad det innebär för hur saker ser ut i minnet.

Variabelnamn och var de “lever” (scope)

Rekursion (svårare rekursion kommer examineras)

Låd-/pil-diagram över minnet

Återigen: det vi har gått igenom på övningarna täcker **inte** hela kursinnehållet

# Uppgifter



# Uppgift 1.1 - scope

Vad kommer detta program skriva ut, och vad händer i datorn?

```
a = 0
```

```
def my_function():  
    print(a)  
    a = 3  
    print(a)
```

```
my_function()  
print(a)
```

# Uppgift 1.2 - scope

Hur kan vi fixa detta program, hur kan vi göra ifall vi vill komma åt den globala variabeln `a` i vår funktion, och ändra den?

```
a = 0
```

```
def my_function():  
    print(a)  
    a = 3  
    print(a)
```

```
my_function()
```



# Uppgift 1.3 - scope

Vad kommer detta program skriva ut, och vad händer i datorn?

```
a = 0
```

```
def my_function():  
    global a  
    a = 3  
    print(a)
```

```
my_function()
```

```
print(a)
```

# Uppgift 1.4 - scope

Viktigt!! Förväxla EJ de tidigare exemplen med tex detta. Vad gör detta?

```
animals = ["katt", "tiger"]
```

```
def my_function():  
    print(animals)
```

```
    animals[1] = "myra"  
    print(animals)
```

```
my_function()
```

```
print(animals)
```

# Uppgift 2- minnesdiagram

Rita en minnesbild med låd- och pildiagram för hur det ser ut då körningen når den kommenterade raden.

```
def funB(a, b):  
    a[1]= "hej"  
    b = ("katt", "lo")  
  
def funA():  
    a = [[1, 2], [3, 4]]  
    b = (5, 6)  
    funB(a, b)  
    #what does the memory look like here?  
    print("this is a", a)  
    print("this is b", b)  
  
funA()
```

# Uppgift 3- minnesdiagram (jävlig)

Rita en minnesbild med låd- och pildiagram för hur det ser ut då körningen når den kommenterade raden. (Jättemånga koncept i denna uppgift!)

```
def funB(t, a):  
    t[1]= "hej"  
    a[0][0]= ["wow"]  
  
def funA():  
    t = ([1, 2], "myra")  
    a = [5, 6]  
    funB(a, t)  
  
    #what does the memory look like here?  
    print("this is a", a)  
    print("this is t", t)  
  
funA()
```

# Uppgift 4- Läsa fil

Visa hur man kan läsa en textfil med ett ord per rad och få orden som en lista i sitt program.

```
['Vad', 'tror', 'du', 'att', 'det', 'hände', 'sen', '?']
```

# Uppgift 5- Förbättra rekursiv fibonacci

Förbättra den rekursiva fibonacci-funktionen så att den inte orsakar en lavin av funktionsanrop. I.e. spara tidigare beräknade värden på något sätt.

Hur skulle vi kunna göra detta?

# Uppgift 6 - Binomialkoefficienter

Beräkna binomialkoefficienter. Gör en lösning som utnyttjar redan beräknade koefficienter och alltså inte orsakar en "lavin" av onödiga funktionsanrop.

Kom ihåg: en binomialkoefficient bestäms av  $n$  och  $k$  genom

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}.$$

Med pascals triangel har vi

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}.$$

Rekursiv :0)

Kan ju lösa denna icke-rekursivt - men vi ska öva på rekursiva funktioner :0)

Basfall?

# Uppgift 7 - Listnivåer

Räkna antalet nästlade listnivåer i en lista, dvs i hur många nivåer listor i listor i listor .... som förekommer. Vi är enbart intresserade av maxdjupet. Alla icke-listor ger 0 som funktionsvärde, t.ex.

5 -> 0,    [3, 4, -1] --> 1,    [3, [4, -3], -9] --> 2    o.s.v.

Vad är basfall(en)?



# Några sista saker

Bra [hemsida](#) :0)

ADK - [DD2350](#) (Obs denna variant, ej att förväxla med fysiks)

Programmering - att skapa något från inget

Håll koll på er GitHub (kommer behöva boka tid tsm med er opponent)

Lycka till!!!111!!  
<3333

Om ni har frågor om ngn uppgift/ngt koncept - så är det bara att maila!

karlhed@kth.se

Repo: <https://github.com/TypAnna/GruProgDD1331>