DD1331

Grundläggande programmering för F1

Föreläsning 1

DD1331 7.5hp
Grundläggande programmering

DD1331 7.5hp
Grundläggande programmering

- Grundläggande programmering i Python

DD1331 7.5hp
Grundläggande programmering

- Grundläggande programmering i Python
- Grundläggande datalogi med exempel i Python

DD1331 7.5hp
Grundläggande programmering

- Grundläggande programmering i Python
- Grundläggande datalogi med exempel i Python

DD1331 7.5hp
Grundläggande programmering

- Grundläggande programmering i Python
- Grundläggande datalogi med exempel i Python
- Examination:

DD1331 7.5hp
Grundläggande programmering

- Grundläggande programmering i Python
- Grundläggande datalogi med exempel i Python
- Examination:

  5 programmeringslabbar att utföra i par 3hp

DD1331 7.5hp
Grundläggande programmering

- Grundläggande programmering i Python
- Grundläggande datalogi med exempel i Python
- Examination:

  5 programmeringslabbar att utföra i par 3hp
  1 programmeringslabb att utföra individuellt 1.5hp

DD1331 7.5hp
Grundläggande programmering

- Grundläggande programmering i Python
- Grundläggande datalogi med exempel i Python
- Examination:

  5 programmeringslabbar att utföra i par 3hp
  1 programmeringslabb att utföra individuellt 1.5hp
  1 skriftlig tentamen 3hp

Grundläggande programmering:

Grundläggande programmering:

- Lagra data (enstaka eller listor)

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)
- Strukturera (funktioner, metoder)

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)
- Strukturera (funktioner, metoder)
- Rekursion

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)
- Strukturera (funktioner, metoder)
- Rekursion

Program ska

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)
- Strukturera (funktioner, metoder)
- Rekursion

Program ska

1.

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)
- Strukturera (funktioner, metoder)
- Rekursion

Program ska

1. Göra rätt

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)
- Strukturera (funktioner, metoder)
- Rekursion

Program ska

1. Göra rätt
2.

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)
- Strukturera (funktioner, metoder)
- Rekursion

Program ska

1. Göra rätt
2. Vara tydliga

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)
- Strukturera (funktioner, metoder)
- Rekursion

Program ska

1. Göra rätt
2. Vara tydliga
3.

Grundläggande programmering:

- Lagra data (enstaka eller listor)
- Repetera, välj
- Kommunicera (indata, utdata)
- Strukturera (funktioner, metoder)
- Rekursion

Program ska

1. Göra rätt
2. Vara tydliga
3. Vara effektiva

Grundläggande datalogi:

Grundläggande datalogi:

- Datastrukturer:

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn

  ADT –

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

Grundläggande datalogi:

- ## Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- ## Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska – t.ex. för sökning och sortering

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska – t.ex. för sökning och sortering
  - Analys

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska – t.ex. för sökning och sortering
  - Analys – hur snabb är algoritmen?

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska – t.ex. för sökning och sortering
  - Analys – hur snabb är algoritmen?
  - Konstruktion

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska – t.ex. för sökning och sortering
  - Analys – hur snabb är algoritmen?
  - Konstruktion – skapa algoritmer för nya problem

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska – t.ex. för sökning och sortering
  - Analys – hur snabb är algoritmen?
  - Konstruktion – skapa algoritmer för nya problem

- Andra aspekter på programmering ...

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska – t.ex. för sökning och sortering
  - Analys – hur snabb är algoritmen?
  - Konstruktion – skapa algoritmer för nya problem

- Andra aspekter på programmering ...

  Tydlighet

Grundläggande datalogi:

- Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska – t.ex. för sökning och sortering
  - Analys – hur snabb är algoritmen?
  - Konstruktion – skapa algoritmer för nya problem

- Andra aspekter på programmering ...

  Tydlighet Stil

Grundläggande datalogi:

- ## Datastrukturer:

  Hur data kan organiseras och lagras i datorn
  ADT – Abstrakta DataTyper

- ## Algoritmer:

  Entydiga beskrivningar av hur problem ska lösas

  - Klassiska – t.ex. för sökning och sortering
  - Analys – hur snabb är algoritmen?
  - Konstruktion – skapa algoritmer för nya problem

- ## Andra aspekter på programmering ...

  Tydlighet Stil Effektivitet

- Program =

- Program = en mängd instruktioner som styr vad datorn ska göra.

- Program = en mängd instruktioner som styr vad datorn ska göra.
- I början, ca 1945 – 1954

- Program = en mängd instruktioner som styr vad datorn ska göra.
- I början, ca 1945 – 1954

  Maskininstruktioner Assemblerspråk

- Program = en mängd instruktioner som styr vad datorn ska göra.

- I början, ca 1945 – 1954

Maskininstruktioner	Assemblerspråk	
10110110110111100010	IN	901
00010110010000101101	STO 99	399
11011011010010010011	IN	901
00100001011101011110	ADD 99	199

- Program = en mängd instruktioner som styr vad datorn ska göra.

- I början, ca 1945 – 1954

Maskininstruktioner	Assemblerspråk	
10110110110111100010	IN	901
00010110010000101101	STO 99	399
11011011010010010011	IN	901
00100001011101011110	ADD 99	199

- Senare 1954 – ...
  Högnivåspråk a = b + c

Programmeringsparadigm = tankemönster för problemlösning

Programmeringsparadigm = tankemönster för problemlösning

- Imperativa

Programmeringsparadigm = tankemönster för problemlösning

- Imperativa
  sekvens av instruktioner
  inklusive val, repetition, underprogram, moduler

Programmeringsparadigm = tankemönster för problemlösning

- Imperativa
  sekvens av instruktioner
  inklusive val, repetition, underprogram, moduler
- Objektorienterade

Programmeringsparadigm = tankemönster för problemlösning

- Imperativa
  sekvens av instruktioner
  inklusive val, repetition, underprogram, moduler
- Objektorienterade
  objekt med egenskaper och aktiviteter

Programmeringsparadigm = tankemönster för problemlösning

- Imperativa
  sekvens av instruktioner
  inklusive val, repetition, underprogram, moduler
- Objektorienterade
  objekt med egenskaper och aktiviteter
- Funktionella

Programmeringsparadigm = tankemönster för problemlösning

- Imperativa
  sekvens av instruktioner
  inklusive val, repetition, underprogram, moduler
- Objektorienterade
  objekt med egenskaper och aktiviteter
- Funktionella
  allt är funktionsdefinitioner

Programmeringsparadigm = tankemönster för problemlösning

- Imperativa
  sekvens av instruktioner
  inklusive val, repetition, underprogram, moduler
- Objektorienterade
  objekt med egenskaper och aktiviteter
- Funktionella
  allt är funktionsdefinitioner
- Logikprogrammering

Programmeringsparadigm = tankemönster för problemlösning

- Imperativa
  sekvens av instruktioner
  inklusive val, repetition, underprogram, moduler
- Objektorienterade
  objekt med egenskaper och aktiviteter
- Funktionella
  allt är funktionsdefinitioner
- Logikprogrammering
- Parallellprogrammering