

Mycket grundläggande in- och utmatning

I denna inledande laboration kommer du att öva på de mest grundläggande operationerna för in- och utmatning.

Mål

Du ska efter denna laboration ha god insikt i

- skillnaden mellan `Get` och `Put` samt `New_Line` och `Skip_Line`
- skillnader och likheter mellan de olika varianterna av `Get` och `Put`
- skillnaderna i beteende vid olika inmatningar till `Get_Line` och kunna hantera dessa

Uppgift

Skriv ett program som hämtar data från tangentbordet enligt exemplen som följer. Det som syns på skärmen när man kör programmet skall stämma till fullo överens med det som står i exemplen. Detta gäller även antal blanktecken mellan ord och tal.

Observera att denna laboration till stor del är till för att introducera er till hur man hanterar vår datormiljö vad det gäller skrivande, kompilering och körning av program. Om ni inte hinner med allt på laborationstiden är det ingen katastrof. Fortsätt på egen tid med resterande delar så att målen med uppgiften är uppfyllda.

På kurswebsidan finns en fil som ni kan hämta med ett litet programskelett som innehåller alla texter som programmet skall skriva ut på skärmen. Detta för att ni skall slippa skriva en massa ren text och på så sätt kunna koncentrera er på hur `Get/Get_Line/Put/Put_Line/New_Line` och `Skip_Line` fungerar.

Ett tips kan vara att ni gör en liten del i taget och ser till att den fungerar för båda de givna körexemplen (se följande sidor). På detta sätt undviker ni att få alla ”fel” på en gång.

OBS! Ditt program skall klara av både körexempel 1 och 2 (samma program). Dock behöver det inte klara av alla möjliga varianter av indata vilket i princip är omöjligt i detta läge.

Programkörningar

Programmet ska ge resultat enligt följande exempel vid olika körningar (användarens indata markeras med *kursiverad* stil). Observera att eventuella inmatningar som är extra skall plockas bort innan nästa inmatning görs.

Observera att du i ditt program endast får ha *en variabel av respektive datatyp* (det får alltså inte finnas t.ex. två stycken strängar). Det blir alltså bara fyra variabler i programmet.

Exempel 1: Skriv in ett heltal: **10**

Du skrev in talet: 10

Skriv in fem heltal: **12 30 27 13 11**

Du skrev in talen: 12 30 27 13 11

Skriv in ett flyttal: **15.20**

Du skrev in flyttalet: 15.200

Skriv in ett heltal och ett flyttal: **67 3.141592**

Du skrev in heltalet: 67

Du skrev in flyttalet: 3.1416

Skriv in ett tecken: **T**

Du skrev in tecknet: T

Skriv in en sträng med 5 tecken: **Kalle**

Du skrev in strängen: Kalle

Skriv in en sträng med 3 tecken: **Kul**

Du skrev in strängen: Kul

Skriv in ett heltal och en sträng med 5 tecken: **-5 Nisse**

Du skrev in talet `|-5|` och strängen `|Nisse|`.

Skriv in en sträng med 3 tecken och ett flyttal: **Ria 1.2**

Du skrev in " 1.200" och "Ria".

Skriv in en sträng som är maximalt 5 tecken: **Bä**

Du skrev in strängen: Bä

Skriv in en sträng som är maximalt 5 tecken: **Nysta**

Du skrev in strängen: Nysta

Exempel 2: Skriv in ett heltal: **14.72**

Du skrev in talet: 14

Skriv in fem heltal: **10000 3860 27 -18** 0 0

Du skrev in talen: 10000 3860 27 -18 0

Skriv in ett flyttal: **15 Nisse**

Du skrev in flyttalet: 15.000

Skriv in ett heltal och ett flyttal: 7

202.718 **Dum text**

Du skrev in heltalet: 7

Du skrev in flyttalet: 202.7180

Skriv in ett tecken: **!K**

Du skrev in tecknet: !

Skriv in en sträng med 5 tecken: **#4?"** .!

Du skrev in strängen: #4?" .

Skriv in en sträng med 3 tecken: **"HA12**

Du skrev in strängen: "HA

Skriv in ett heltal och en sträng med 5 tecken: **1 Ploja!**

Du skrev in talet |1| och strängen |Ploja|.

Skriv in en sträng med 3 tecken och ett flyttal: **Led**

012.3456.23

Du skrev in "12.346" och "Led".

Skriv in en sträng som är maximalt 5 tecken: **Per är bra**

Du skrev in strängen: Per ä

Skriv in en sträng som är maximalt 5 tecken: **KUL**

Du skrev in strängen: KUL

2010-09-06