

Dec 18, 17 11:42

snabbstartsmaterial.txt

Page 1/3

FÖRELÄSNING 0: Introduktion av Ada
=====

OBS! Detta är antagligen det enda papper som delas ut under hela FÖ-serien. Det finns förutom detta ett antal OH:n som kommer att användas under nästa FÖ, men sen är det endast tavelskrivande och diskussioner som gäller.

Vad kommer vi att behöva så här i kursstarten? Lite begrepp och lite problemlösning. Dessutom lite hur ett program ser ut. Vi börjar med lite begrepp.

Ett antal "reserverade identifierare" (reserverade ord). Dessa används för att strukturera upp programmet och får bara användas till just detta. Lite "ihopklumpande" efter hur/när man använder dem:

procedure ... is	loop
begin	end loop;
end ...;	
with ...;	while ... loop
use ...;	end loop;
	for ... in ... loop
if ... then	end loop;
elsif ... then	
else	exit;
end if;	exit when ...;

Ett antal "datatyper" ("beskrivningar" av hur data ser ut). Dessa är några av de "fördefinierade identifierare" vi kommer att komma i kontakt med:

Integer	= heltal (ex. 1, 2, 145, -4)
Float	= flyttal (ex. 1.0, 2.0, 3.14, -3.0)
Character	= tecken (ex. 'a', 'b', '2', '#')
String(...)	= "textsnutt" (ex. "kalle", "berta")
Boolean	= sanningsvärde (False, True)

Ett antal "variabler" (lagringsutrymmen att lägga data i). Dessa blir såkallade "egendefinierade identifierare":

```
I : Integer;
F : Float;
C : Character;
S : String(1 .. 5);
B : Boolean;
```

När man skapar dessa variabler använder man datatyperna för att tala om vilken sorts variabel det är.

OBS! En identifierare identifierar något i programmet och skall därför vara unikt (inom det "område i programmet" som använder denna).

Regel för identifierare:

En "identifierare" måste inledas med en bokstav. Därefter får det följa fler bokstäver och/eller siffror och/eller understrykningsstreck. Man får dock inte avsluta identifieraren med ett understrykningsstreck eller ha två sådana i rad inuti identifieraren.

Dec 18, 17 11:42

snabbstartsmaterial.txt

Page 2/3

Det som är beskrivet hittills är saker som redan finns i Ada utan att man behöver säga något extra. När man kommer till programmeringen behöver man dock ofta lite mer att "leka med".

Det som behövs kan vara matematiska bibliotek med "Cos", "Sin", etc. eller hantering av externa enheter såsom tangentbord och skärm (såkallad in- och utmatning vilket kallas IO efter det engelska "In and Out"). Dessa delar har man i Ada delat upp i olika moduler kallade "paket". Det finns många sådana fördefinierade och här kommer ett par stycken med några av de saker som finns inuti:

Ada.Text_IO

```
Put(...);
Put_Line(...);
New_Line;
New_Line(...);

Get(...);
Get_Line(...);
Skip_Line;
Skip_Line(...);
```

Ada.Float_Text_IO

```
Put(...);

Get(...);
```

Ada.Integer_Text_IO

```
Put(...);

Get(...);
```

Ada.Numerics.Elementary_Functions

Sqrt(...)	Log(...)	Exp(...)
Sin(...)	Arcsin(...)	
Cos(...)	Arccos(...)	
Tan(...)	Arctan(...)	
Cot(...)	Arccot(...)	
Sinh(...)	Arcsinh(...)	
Cosh(...)	Arccosh(...)	
Tanh(...)	Arctanh(...)	
Coth(...)	Arccoth(...)	

OBS! Det är inte något ';' efter de saker som finns inuti det sista paketet. Varför kommer vi tillbaka till senare på den FÖ som handlar om underprogram.

FÖRELÄSNING 0: Introduktion av Ada

=====

(Föreläsarens material: Programexempel)

Hur skriver vi nu ett program med hjälp av dessa saker? Vi kan börja med ett exempel från labserien (den första introduktionslaborationen som följer efter denna FÖ i denna kurs). Följande dialog vill vi att den skall utföras med datorn:

```
Mata in ett heltal: 10
Du matade in heltalet: 10
```

```
Mata in en sträng med exakt 5 tecken och ett heltal: Kalle 27
Du matade in heltalet 27 och strängen "Kalle".
```

OBS! Det är meningen att datorn (ditt program) skall skriva ut allt i dialogen på skärmen UTOM det som står efter frågorna "Mata in ...:". Dessa data skall användaren mata in på tangentbordet och det är just dessa inmatningar som sen skall skrivas ut på raden efter (av ditt program).

Vi börjar med uppgiften på tavlan (bra att tala om hur många rader det blir och vilka rader man använder när man skriver programmet):

```
1      -- with Ada.Text_IO;          -- use Ada.Text_IO;
2      -- with Ada.Integer_Text_IO;  -- use Ada.Integer_Text_IO;
3      --
4      procedure In_Och_Utmatning is
5          --
6          -- I : Integer;
7          -- -- S : String(1 .. 5):
8          --
9      begin
10         -- Ada.Text_IO.Put("Mata in ett heltal: ");
11         -- Ada.Integer_Text_IO.Get(I);
12         -- -- Skip_Line
13         -- Ada.Text_IO.Put("Du matade in heltalet: ");
14         -- Ada.Integer_Text_IO.Put(I);
15         -- -- Ada.Integer_Text_IO.Put(I, Width => 0);
16         -- Ada.Text_IO.New_Line;
17         --
18         -- Put("Mata in en sträng med exakt 5 tecken och ett heltal: ");
19         -- Get(S);
20         -- Get(I);
21         -- Skip_Line;
22         -- Put("Du matade in heltalet ");
23         -- Put(I, Width => 0);
24         -- Put(" och strängen """);
25         -- Put(S);
26         -- Put_Line("".");
27     end In_Och_Utmatning;
```