

# Handbók fyrir mofó

Baldur Þór Emilsson

9. maí 2010

## Útdráttur

Útdráttur...

## Efnisyfirlit

<b>1</b>	<b>Inngangur</b>	<b>2</b>
<b>2</b>	<b>Notkun og uppsetning</b>	<b>2</b>
<b>3</b>	<b>Málfræði</b>	<b>4</b>
3.1	Frumeiningar málsins . . . . .	4
3.1.1	Athugasemdir . . . . .	4
3.1.2	Lykilorð . . . . .	4
3.2	Mállýsing . . . . .	4
<b>4</b>	<b>Merking málsins</b>	<b>4</b>
4.1	Einingar og einingaraðgerðir . . . . .	4
4.2	Gildi . . . . .	4
4.3	Breytur . . . . .	4
4.4	Merking segða . . . . .	4
4.4.1	Heiltölusegð . . . . .	4
4.4.2	Fleytitölusegð . . . . .	4
4.4.3	Stafsegð . . . . .	4
4.4.4	Strengsegð . . . . .	4
4.4.5	Listasegð . . . . .	4
4.4.6	return-segð . . . . .	4
4.4.7	Röksegðir . . . . .	4
4.4.8	Kallsegð . . . . .	4
4.4.9	Tvíundaraðgerðir . . . . .	4

4.4.10	Einundaraðgerðir . . . . .	4
4.4.11	if-segð . . . . .	4
4.4.12	while-segð . . . . .	4
<b>5</b>	<b>Til hliðsjónar ... hendið þessu eftir notkun</b>	<b>4</b>
5.1	Mállýsing með syntax pakkanum í LaTeX . . . . .	4
5.2	Ýmis málrit með hjálp syntax pakkans í LaTeX . . . . .	5
5.2.1	Program . . . . .	5
5.2.2	Modules . . . . .	5
5.2.3	Stofnar . . . . .	6
5.2.4	Skilgreiningar . . . . .	6
5.2.5	Segðir . . . . .	6
5.2.6	Annar möguleiki . . . . .	7

## 1 Inngangur

Mofo er forritunarmál sem þróað var í áfanganum Þýðendur í Háskóla Íslands vorið 2010. Það byggir á málinu Morpho (<http://morpho.cs.hi.is>) eftir Snorra Agnarsson, en sækir einnig nokkuð úr hugmyndafræði Python. Mofo er mjög frumstætt og takmarkað þegar að útfærslumöguleikum kemur, en tenging þess í Morpho eykur notagildi þess til muna.

## 2 Notkun og uppsetning

Grunnkóða þýðandans má nálgast á <http://www.baldur.biz/mofo/mofolang.zip> og með honum fylgja Bash skriftur fyrir þýðanda og keyrslumhverfi. Í zip skránni er einnig að finna umhverfi fyrir Morpho, svo notandi þarf einungis Java til að keyra Mofo forrit.

Mofo notar Flex og Bison í þýðandanum og er skrifað í C++. Til að þýða hann á Linux er hægt að nota `make` skipunina.

Mofo kóði er geymdur í skrá sem enda á `.mof`.



## **3 Málfræði**

### **3.1 Frumeiningar málsins**

#### **3.1.1 Athugasemdir**

#### **3.1.2 Lykilorð**

### **3.2 Mállýsing**

## **4 Merking málsins**

### **4.1 Einingar og einingaraðgerðir**

### **4.2 Gildi**

### **4.3 Breytur**

### **4.4 Merking segða**

#### **4.4.1 Heiltölusegð**

#### **4.4.2 Fleytitölusegð**

#### **4.4.3 Stafsegð**

#### **4.4.4 Strengsegð**

#### **4.4.5 Listasegð**

#### **4.4.6 return-segð**

#### **4.4.7 Röksegðir**

#### **4.4.8 Kallsegð**

#### **4.4.9 Tvíundaraðgerðir**

#### **4.4.10 Einundaraðgerðir**

#### **4.4.11 if-segð**

#### **4.4.12 while-segð**

## **5 Til hliðsjónar ... hendið þessu eftir notkun**

### **5.1 Mállýsing með syntax pakkanum í LaTeX**

Hér er eilítill mállýsing sniðsett með hjálp `syntax` pakkans sem nota má í `LaTeX`. Þið getið notað þessa aðferð, ef það hentar ykkur, til að skilgeina ykkar forritun-

$$\langle x \rangle ::= \langle x \rangle \langle y \rangle$$
$$| \langle y \rangle \text{ 'x' }$$

Hér eru nokkur málrit skilgreind með hjálp `syntax` þakans. Breytið þessu eftir þörfum, eða notið aðrar aðferðir ef það hentar ykkur betur.

Takið eftir að ráðlegt er að gera einhverjar breytingar hér, þó ekki væri til annars en að gera textalýsingar á merkingu einfaldari með tilliti til forgangs ýmissa mál fyrirbæra.

Diagram illustrating the structure of a module declaration:

```

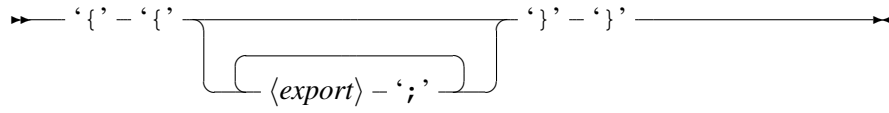
{
  <string> = <module> ;
}

```

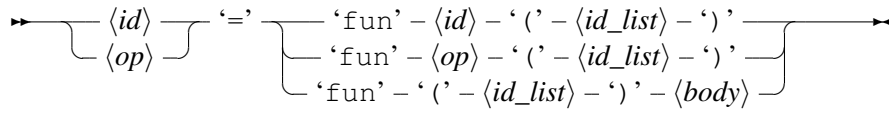
The diagram shows the components of the declaration: `<string>`, `=`, `<module>`, and `;`. Brackets indicate the grouping of these components into a single declaration.

5

$\langle \text{textmodule} \rangle$ :

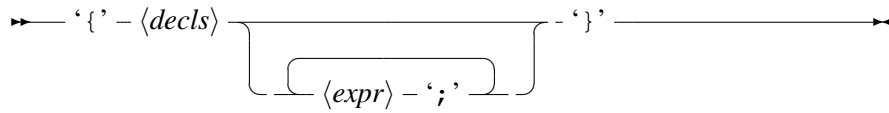


$\langle \text{export} \rangle$ :



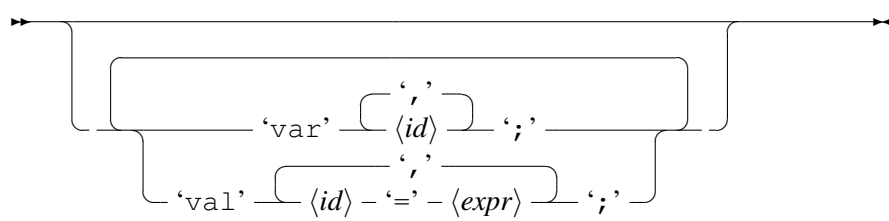
### 5.2.3 Stofnar

$\langle \text{body} \rangle$ :



### 5.2.4 Skilgreiningar

$\langle \text{decls} \rangle$ :

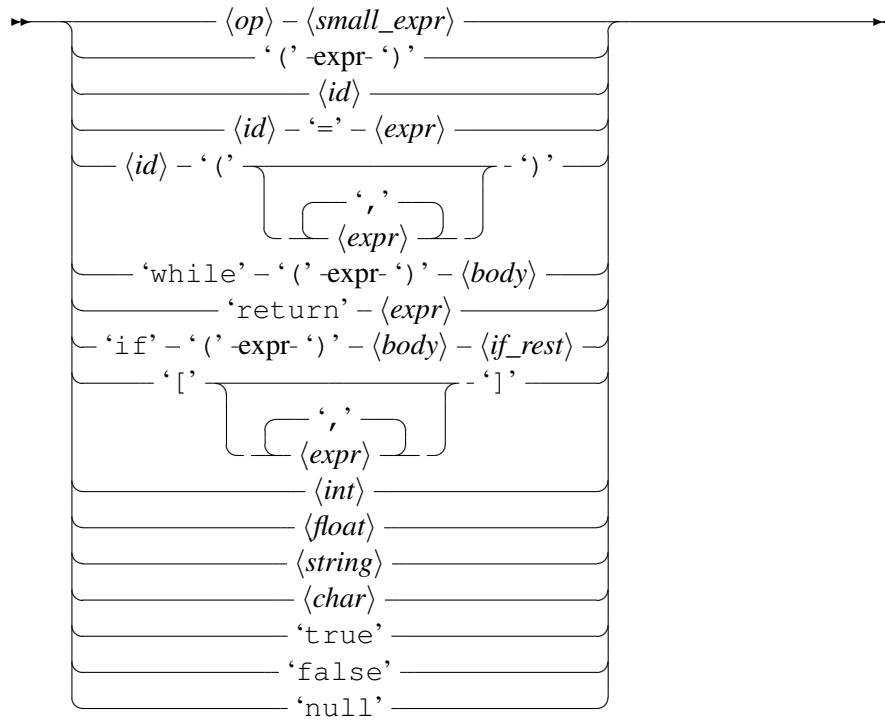


### 5.2.5 Segðir

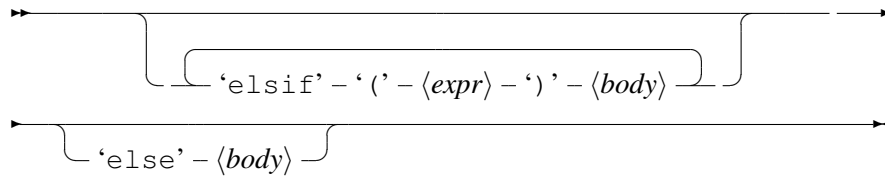
$\langle \text{expr} \rangle$ :



$\langle small\_expr \rangle$ :

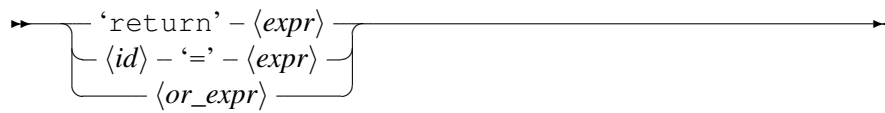


$\langle if\_rest \rangle$ :



## 5.2.6 Annar möguleiki

$\langle expr \rangle$ :



$\langle or\_expr \rangle$ :



$\langle and\_expr \rangle$ :

Diagram illustrating the structure of  $\langle and\_expr \rangle$ . It shows a sequence starting with  $\langle not\_expr \rangle$ , followed by the operator  $\& \&$ , and then continuing with the rest of the expression.

$\langle not\_expr \rangle$ :

$\langle binop\_expr \rangle$

$$\langle binop\_expr \rangle:$$

$$\begin{array}{c} \text{---} \langle op \rangle \text{---} \\ \text{---} \langle small\_expr \rangle \text{---} \end{array}$$

$\langle \text{small\_expr} \rangle$ :

- $\langle op \rangle - \langle \text{small\_expr} \rangle$
- $'(' - \text{expr} - ')'$
- $\langle id \rangle$
- $'(' - \langle \text{expr} \rangle - ', \rangle$
- $\langle \text{while} \rangle - '(' - \text{expr} - ')' - \langle body \rangle$
- $\langle \text{if} \rangle - '(' - \text{expr} - ')' - \langle body \rangle - \langle \text{if\_rest} \rangle$
- $\langle '[' \rangle - \langle \text{expr} \rangle - \langle ',' \rangle - \langle ']' \rangle$
- $\langle int \rangle$
- $\langle float \rangle$
- $\langle string \rangle$
- $\langle char \rangle$
- $\langle true \rangle$
- $\langle false \rangle$
- $\langle null \rangle$