

Handbók fyrir Mofo

Baldur Þór Emilsson

10. maí 2010



Útdráttur

Mofo er einföld útgáfa af forritunarmálinu Morpho. Það er mjög takmarkað en tenging þess við Morpho eykur notagildi þess til muna.

Efnisyfirlit

1 Innangangur

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 í hugmyndafræði Python.

Mofo er mjög frumstætt og takmarkað þegar að útfærslumöguleikum kemur, en tenging þess við Morpho eykur notagildi þess til muna.

2 Notkun og uppsetning

Grunnkóða þýðandans má nálgast á <http://notendur.hi.is/bthe1/mofo/mofolang.zip> og með honum fylgja Bash skriftur fyrir þýðanda og keyrsluumhverfi. Í zip skránni er einnig að finna umhverfi fyrir Morpho, svo notandi þarf einungis Java til að keyra Mofo forrit. Java þarf að vera aðgengilegt í PATH í skelinni.

Mofo notar Flex lesgreini og Bison þáttara og er þýðandinn skrifaður í C++. Til að þýða hann á Linux er hægt að nota `make` skipunina. Hún virkar eflaust líka á öðrun Unix stýrikerfum.

Mofo kóði er geymdur í skráum sem enda á `.mof`. Þær eru þýddar með skipuninni `./mofocompile test` fyrir skrána `test.mof`. Þegar skráin hefur verið þýdd er hægt að keyra hana með `./moforun test`. Þetta á þó aðeins við um skrár sem hafa `main`-fall, en meira um það síðar.

3 Málfræði

3.1 Frumeiningar málsins

3.1.1 Athugasemdir

Engar athugasemdir eru útfærðar í Mofo.

3.1.2 Lykilorð

Lykilorðin eru `def`, `pass`, `if`, `else`, `while`, `break`, `continue`, `return`, `and`, `or`, `true`, `false`, `not`, `null`, `endmodule`.

3.2 Sértákn

Í Mofo eru (og) sértákn sem afmarka viðfangslista í föllum. Einnig má nota þá í reiknisegðum til að hækka forgang.

Hreiðrun í Mofo er táknuð með inndrætti, áður en hann hefst er táknið : notað til að skilgreina að hreiðrun sé að hefjast.

Stök í viðfangslistum falla eru aðgreind með kommu: , . Þetta á bæði við um skilgreiningu falls og kallsegðir.

3.3 Inndráttur

Stofn falla, if-segða og while-segða er skilgreindur með inndrætti. Fjögur stafabil tákna einn inndrátt, og má hann hvorki vera meiri né minni. Einnig er ekki hægt að nota tab-staf fyrir inndrátt. Þegar stofni lýkur færast inndrátturinn aftur á fyrra stig.

Ekki þarf að passa upp á inndrátt í tómun línum. Þær mega innihalda eins mörg stafabil og tab-stafi eins og forritaranum þykir þurfa. Eins mega vera auðar línur á milli hverra annarra lína í málinu þar sem það þykir henta fyrir læsileika eða annað.

3.4 Mállýsing

3.4.1 Grunneiningar

$\langle string \rangle$, $\langle char \rangle$, $\langle int \rangle$, $\langle double \rangle$ og $\langle opname \rangle$ eru eins og í Morpho.

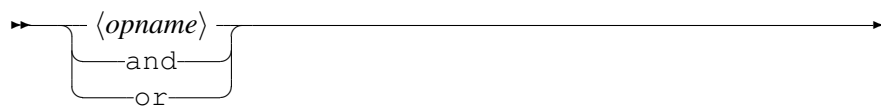
$\langle name \rangle$:

→ $\left('a-zA-Z' \right)$ →
 $\left('a-zA-Z0-9_'' \right)$

$\langle literal \rangle$:



$\langle binop \rangle$:

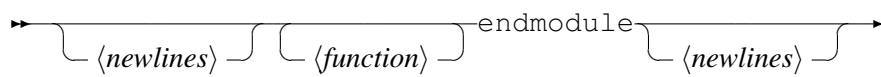


$\langle newlines \rangle$:

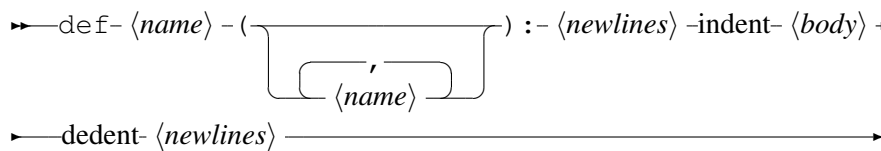


3.4.2 Forrit

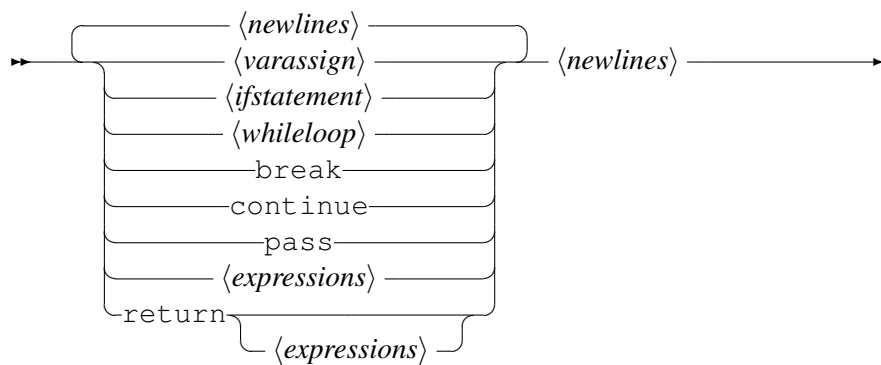
$\langle program \rangle$:



$\langle function \rangle$:



$\langle body \rangle$:



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

$\rightarrow \langle \text{name} \rangle \text{ :- } \langle \text{expressions} \rangle \rightarrow$

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

$\rightarrow \text{if } \langle \text{expressions} \rangle \text{ : } \langle \text{newlines} \rangle \text{ indent } \langle \text{body} \rangle \text{ dedent } \rightarrow$

$\rightarrow \text{else } \langle \text{expressions} \rangle \text{ :- } \langle \text{newlines} \rangle \text{ indent } \langle \text{body} \rangle \text{ dedent } \rightarrow$

$\rightarrow \text{else :- } \langle \text{newlines} \rangle \text{ indent } \langle \text{body} \rangle \text{ dedent } \rightarrow$

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

$\rightarrow \text{while } \langle \text{epxpressions} \rangle \text{ :- } \langle \text{newlines} \rangle \text{ indent } \langle \text{body} \rangle \text{ dedent } \rightarrow$

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

$\rightarrow \langle \text{binop} \rangle \langle \text{expression} \rangle \rightarrow$

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

$\rightarrow \text{not } \langle \text{expression} \rangle \rightarrow$

$\rightarrow (\langle \text{expressions} \rangle) \rightarrow$

$\rightarrow \langle \text{name} \rangle \rightarrow$

$\rightarrow \langle \text{functioncall} \rangle \rightarrow$

$\rightarrow \langle \text{literal} \rangle \rightarrow$

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

$\rightarrow \langle \text{name} \rangle (\langle \text{expressions} \rangle) \rightarrow$

4 Merking málsins

4.1 Einingar og einingaraðgerðir

Hver Mofo kóðaskrá samsvarar einni einingu í Morpho. Engin útfærsla er á tengingu við aðrar einingar aðrar en BASIS í Morpho, sem er sjálfgefin. Mofo skrár eru þó þýddar yfir í Morpho og hægt er að nota þær einingar sem úr Mofo koma í Morpho, þó notagildið sé takmarkað þar sem allar einingarnar eru tengdar við BASIS.

Mofo kóðaskrá sem inniheldur `main` fall er þýdd yfir í keyrsluhæfa Morpho skrá (`.mexe`), en sé það ekki til staðar þýðist kóðinn yfir í `.mmod` skrá sem hægt er að tengja við Morpho.

4.2 Gildi

Öll gildi í Mofo (heiltölur, fleytitölur, strengir, stafsegðir, `true`, `false` og `null`) varpast yfir í samsvarandi gildi í Morpho.

4.2.1 Sanngildi

Segðir sem lesa sanngildi úr reiknisegð lesa `false`, `null` og tölugildið 0 sem ósatt, allt annað er satt.

4.3 Breytur

Ekki þarf að skilgreina breytur sérstaklega áður en þeim er gefið gildi. Engin tögun er heldur í Mofo, svo sama breyta getur innihaldið gildi af mismunandi tagi á mismunandi tíma.

Dæmi um gildisveitingu er $x = y$, þar sem x er breytunafnið og y er reiknisegð.

4.4 Merking segða

Heiltölur, fleytitölur, strengir, stafsegðir, `true`, `false` og `null` virka eins og í Morpho.

4.4.1 pass-segðir

Í Mofo er `pass` segðin svipuð NOOP í smalamáli og stakri semíkommu í Morpho. Hinsvegar er hún hunsuð við þýðingu svo engin aðgerð er framkvæmd af tölvunni þar sem hún kemur fyrir.

4.4.2 Reiknisegðir

Reiknisegðir eru þær segðir sem skila gildi í Mofo. Þær innihalda breytur, gildi, röksegðir, kallsegðir og tvíundarsegðir. Þegar allir liðir reiknisegðar hafa verið framkvæmdir fæst gildi hennar (nema þegar hoppað er yfir liði, sjá Röksegðir). Það gildi má nota við gildisveitingu breyta, sem viðfang í kallsegð eða sem viðfang í if- og while-segðir.

4.4.3 Listasegð

Engin listasegð er í Mofo, og ekki er hægt að byggja lista í því þar sem tvíundaraðgerðin sem notuð er til þess í Morpho er ekki aðgengileg úr Mofo. Ástæður þess eru raknar í hlutanum um tvíundaraðgerðir.

4.4.4 return-segð

Hægt er að nota lykilorðið `return` til að ljúka keyrslu falls áður en stofni þess er lokið. Bæði getur það staðið eitt og sér, en einnig getur reiknisegð fylgt því í sömu línu, og þá lýkur keyrslu fallsins með þeirri segð.

4.4.5 Röksegðir

Röksegðir í Mofo skila annað hvort `true` eða `false` gildum. Tvíundaraðgerðirnar `and` og `or` eru útfærðar þar sem `and` hefur hærri forgang, og eru þær báðar vinstri tengdar. Prefix einundaraðgerðin `not` er einnig útfærð, og hefur hún hæstan forgang.

Ef vinstri hlið `or` segðar hefur sanngildið `true` er hægri hlið hennar ekki keyrð. Á sama hátt er hægri hlið `and` segðar ekki keyrð ef sanngildi vinstri segðar hennar er `false`.

4.4.6 Kallsegð

Kalla má á föll í reiknisegðum. Skilagildi þeirra er gildið sem fæst úr síðustu segð þeirra, og ef aðeins eru `pass` segðir í stofni fallsins er skilagildi þess óskilgreint. Viðföng falls eru reiknisegðir, svo stef á borð við `f(g())` og `f(a++b, c and d)` eru leyfileg.

4.4.7 Tvíundaraðgerðir

Tvíundaraðgerðir Mofo eru svipaðar tvíundaraðgerðum Morpho. Munurinn liggur helst í röksegðunum `and` og `or`, sem koma í stað segðanna `&&` og `||`, og í þeirri breytingu að segðin `:` er ekki leyfileg segð í Mofo. Það skýrst af því að

táknið hefur málfræðilega merkingu og er því frátekið (sjá hlutann um sértákn að ofan).

Tenging og forgangur er eins og í Morpho.

4.4.8 Einundaraðgerðir

Aðeins ein einundarsegð er útfærð í Mofo: `not`. Henni er lýst í hlutanum um röksegðir hér að ofan.

4.4.9 if-segð

Í Mofo má nota if-segðir til að stýra keyrslu forrits. Þær hefjast á forminu `if x:` þar sem `x` er reiknisegð. Ef gildi hennar hefur sanngildið `true` er stofn segðarinnar keyrður.

Ef sanngildið er `false` er stofninum sleppt, en möguleiki er á else-hluta þar á eftir. Hann getur einnig haft reiknisegð, og er þá á forminu `else x:`. Raða má eins mörgum else-hlutum neðan við if-segð eins og þurfa þykir. Fyrsti else-hlutinn sem hefur reiknisegð með sanngildið `true` er keyrður, og restinni sleppt. Bæta má við einum else-hluta neðst sem enga reiknisegð hefur, og er hann þá keyrður ef bæði if-hlutinn og öllum hinum else-hlutunum var sleppt.

4.4.10 while-segð

Mofo hefur while-segðir, og virka þær svipað og í Morpho. Þær eru á sniðinu `while x:` þar sem `x` er reiknisegð sem framkvæmd í hvert sinn sem ítra á í gegnum stofn while-segðarinnar. Ítrað er í gegnum while-segðina þangað til reiknisegðin hefur sannleiksgildið `false`.

Nota má `break` til að hætta keyrslu while-segðar og halda áfram að keyra forritið frá þeim stað þar sem stofni hennar lýkur, og `continue` segð til að hefja næstu ítrun áður en núverandi ítrun er lokið.