

HÁSKÓLI ÍSLANDS

HÖGNI FREYR GUNNARSSON, SON VAN NGUYEN, VIKTOR ÞÓR FREYSSON

Handbók fyrir Nanomorpho

21. apríl 2020



HÁSKÓLI ÍSLANDS

Nanomorpho er einfalt forritunarmál sem tekur eftir Morpho málinu. Það er bálkmótað hvað föll varðar og styður bara staðværar breytur. Þar sem Nanomorpho er byggt á Morpho og Morpho er byggt ofan á Java virkar Nanomorpho sem minni og léttari útgáfa af Morpho. Ekki eru allir sömu möguleikar til staðar í Nanomorpho miðað við Morpho. Nanomorpho keyrir á Morpho sýndarvélinni.

Efnisyfirlit

1	Inngangur	3
2	Notkun og uppsetning	4
3	Málfræði	4
3.1	Frumeiningar málsins	4
3.1.1	Athugasemdir	4
3.1.2	Lykilorð	5
3.2	Mállýsing	5
3.2.1	Forrit	5
3.2.2	Föll	5
3.2.3	Skilgreiningar	5
3.2.4	Segðir	6
3.2.5	Stofnar	6
4	Merking málsins	6
4.1	Gildi	6
4.2	Breytur	6
4.3	Merking segða	7
4.3.1	null-segð	7
4.3.2	Boolean segðir	7
4.3.3	Heiltölusegð	8
4.3.4	Fleytitölusegð	8
4.3.5	Stafsegð	9
4.3.6	Strengsegð	9
4.3.7	return-segð	9
4.3.8	Röksegðir	9
4.3.9	Kallsegð	9
4.3.10	Tvíundaraðgerðir	9
4.3.11	Einundaraðgerðir	10
4.3.12	if-segð	10
4.3.13	while-segð	10
4.4	Föll og forrit	11

1 Inngangur

Nanomorpho er forritunarmál sem þýðist yfir í Morpho smalamál. Nanomorpho þýðandinn er skrifaður í Java með notkun JFlex (e. lexical analyzer generator). Nanomorpho er með einfalt módula kerfi sem er hannað með það í huga að vera einfalt og þægilegt í notkun. Morpho.jar skráin þarf að vera til staðar til að keyra Nanomorpho forritsbúta.

Dæmi um einfaldan forritsbúta í Nanomorpho:

```
//Example of a comment
//Simple test function in Nanomorpho
test()
{
    var a,b,c;
    a = 2+3;
    b = "HelloWorld";
    c = 2.5;

    if(a == 5) {
        writeln(a);
        writeln(b);
        writeln(c);
    };

    if(a != 5) {
        writeln("if a is != 5");
    }
    elsif (a < 5) {
        writeln("else if a is < 5");
    }
    else {
        writeln("else a is >= 5");
    };

    return "Return value is " ++ 1;
}
```

Nanomorpho keyrir á Morpho sýndarvélinni og getur þ.a.l. nýtt sér öll innbyggðu föll sem má finna í BASIS einingunni úr Morpho.

2 Notkun og uppsetning

Til að keyra og þýða forrit skrifuð í NanoMorpho er mælt með að sækja allar skrár af

<https://github.com/Viktorthor/Compiler-HSV>

Í þeirri möppu er að finna skrána `test.s` og inniheldur skráin keyrsluhæft prufuforrit skrifað í NanoMorpho. Sömuleiðis er dæmi um kóðann á lokablaðsíðu handbókarinnar. Ráðlagt er að sækja alla möppuna á github og svo er hægt að þýða skrána og keyra á tvennskunar hátt.

```
java -jar jflex-full-1.7.0.jar nanoMorpho.jflex
```

Þá verður til `.mexe` skrá sem inniheldur jafngildu Morpho smalamáli og var til staðar fyrir þýðingu.

```
javac Lexer.java Parser.java ParserVal.java
```

Smalamálið er svo þýtt með Morpho sýndarvélinni með eftirfarandi skipun:

```
java Parser test.s | java -jar morpho.jar -c
```

Svo að lokum er hægt að keyra skrána með eftirfarandi skipun:

```
java -jar morpho.jar test
```

Einnig til að gera hlutina þægilegri er einfaldlega hægt að keyra eftirfarandi skipun og skráin `test.s` verður þýdd og keyrð fyrir þig.

```
make runsimpletest
```

3 Málfræði

Málfræði Nanomorpho, sem er undirtegund Morpho, minnir helst á C, C++, C# og Java.

3.1 Frumeiningar málsins

3.1.1 Athugasemdir

Athugasemdir má setja hvert sem er þar sem er leyfilegt að hafa 'white-space'. Þ.e.a.s milli eininga í málinu. Hægt er að setja athugasemd í línur eða fyrir aftan kóða.

Línu athugasemd byrjar á tveimur skástrikum `//'` og gildir út línuna. Allt eftir tveimur skástrikum telst með sem athugasemd.

3.1.2 Lykilorð

Nanomorpho hefur eftirfarandi lykilorð:

`'if', 'elsif', 'else', 'var', 'while', 'return'`

3.2 Mállýsing

Hér að neðan má sjá málrit sem lýsa NanoMorpho.

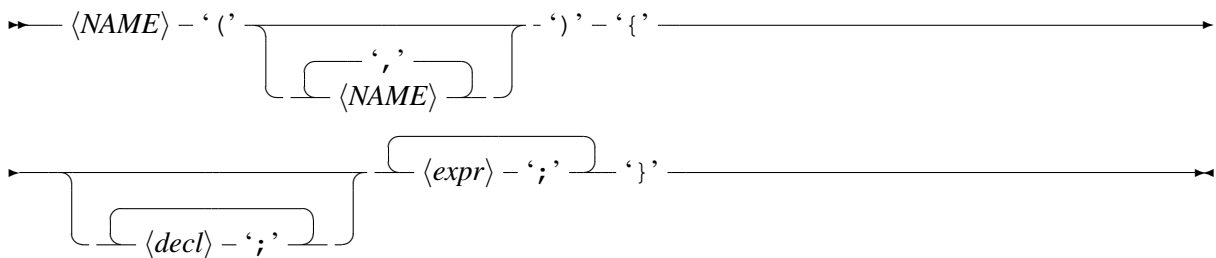
3.2.1 Forrit

$\langle program \rangle$:



3.2.2 Föll

$\langle function \rangle$:



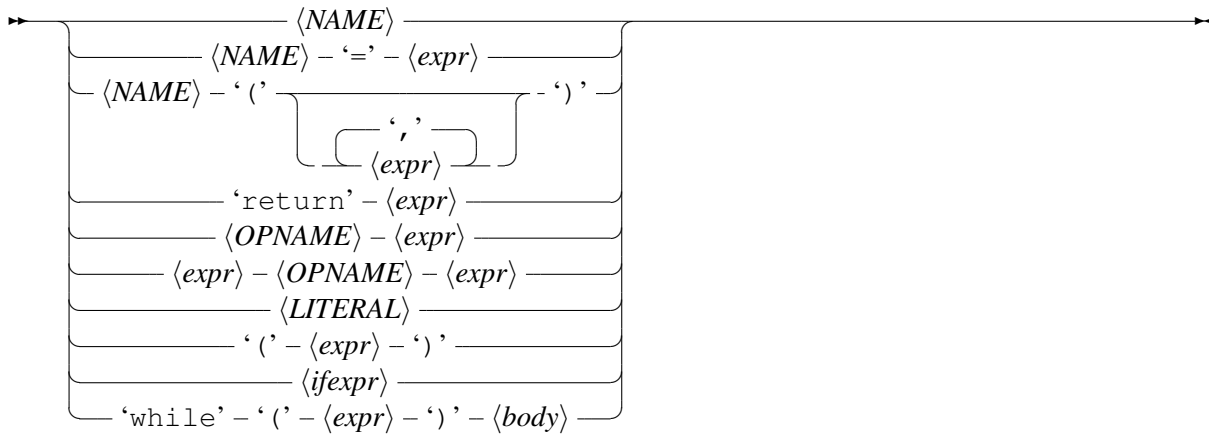
3.2.3 Skilgreiningar

$\langle decl \rangle$:

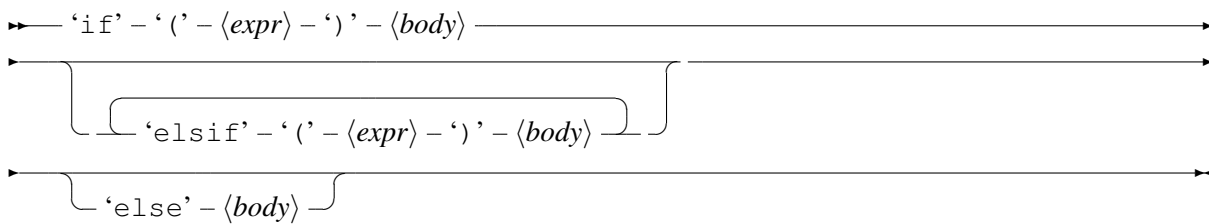


3.2.4 Segðir

$\langle expr \rangle$:



$\langle ifexpr \rangle$:



3.2.5 Stofnar

$\langle body \rangle$:



4 Merking málsins

4.1 Gildi

Gildi í NanoMorpho geta verið heiltölur, fleytitölur, strengir, true, false og null.

4.2 Breytur

Breytur geta tekið gildin sem eru talin upp hér að ofan.

Nýjar breytur eru skilgreindar á eftirfarandi hátt:

```
var a;  
a = <expr>;
```

Þar sem <expr> er lögleg segð úr málinu, hægt er að sjá lýsingar á löglegum segðum með málríti í kafla 3.2.4

Til að skilgreina margar nýjar breytur þarf að gera það í sömu línu fyrir hvert fall. Þ.e.a.s:

```
var x, y, z;  
x = <expr>;  
y = <expr>;  
z = <expr>;
```

4.3 Merking segða

4.3.1 null-segð

Null-segð í NanoMorpho er á sama formi og í Morpho & Java og er merkt með `null`.

4.3.2 Boolean segðir

Segðirnar `true` og `false` eru boolean segðir. Gildi segðana er sú sama og Java Boolean object

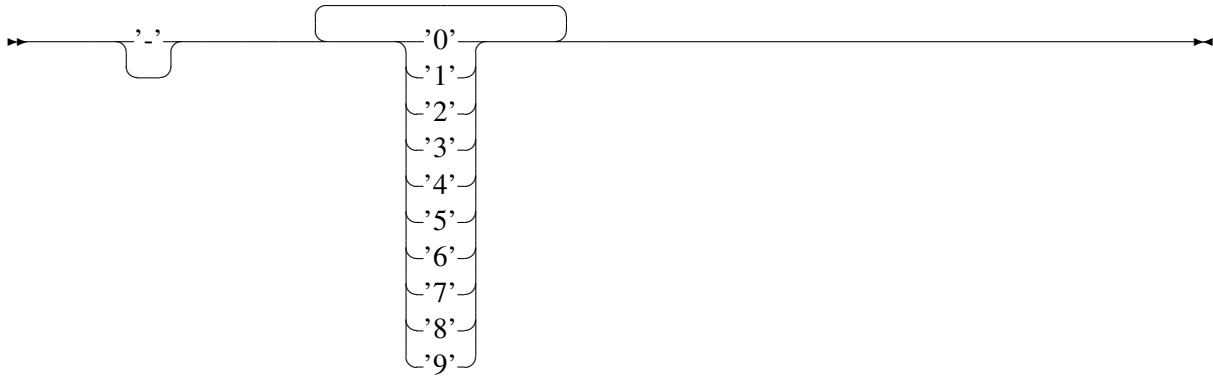
4.3.3 Heiltölusegð

Heiltölusegð í NanoMorpho er til hagna eins og heiltölusegð (e. *Integer*) í Morpho, þ.e.a.s. hún er einhver tala n , þar sem:

$$-2,147,483,648 \leq n \leq 2,147,483,647$$

Málgit fyrir heiltölusegð er:

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



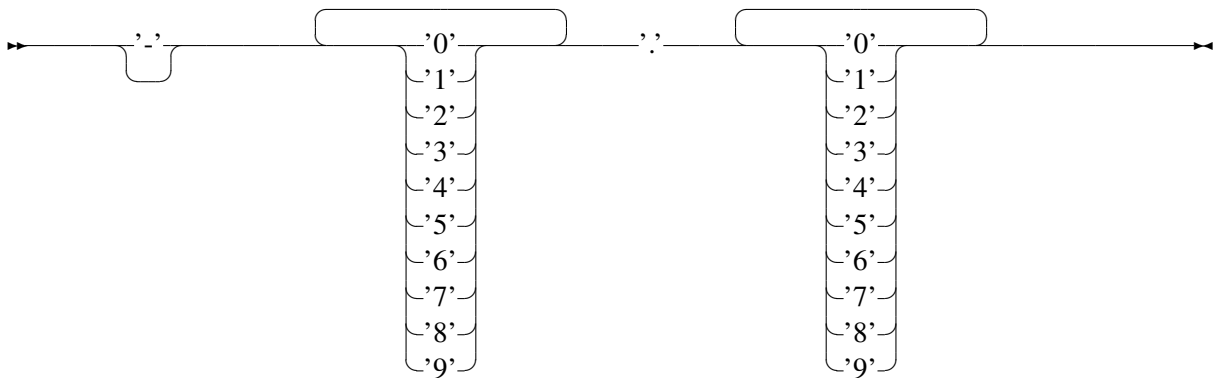
4.3.4 Fleytitölusegð

Fleytitölusegðir í NanoMorpho er einhver tala n , þar sem:

$$-\infty < n < \infty.$$

Málgit fyrir fleytitölusegð er:

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



4.3.5 Stafsegð

Stafsegð í NanoMorpho er haldið til hagana sbr. Morpho stafsegðar hlut. Þ.e.a.s hún er afmörkuð með táknuinu '.

4.3.6 Strengsegð

Strengsegð í NanoMorpho er eins og í Morpho. Þ.e.a.s. hún er afmörkuð með táknuinu ".

4.3.7 return-segð

Return-segð í NanoMorpho er á forminu `return <expr>`. Segðin getur einungis komið fyrir innan stofns sérhvers falls. Þegar segðin kemur fyrir í keyrslu er hægri hlið `<expr>` reiknuð út og sett sem skilagildi fallsins.

4.3.8 Röksegðir

Röksegðir í NanoMorpho eru settar upp eins og röksegðir í Morpho.

Táknin fyrir samanburðavirkja eru eftirfarandi:

`<, >, ≤, ≥, og ==`

Rökvirkinn `not` er settur með hefðbundnum hætti með upphrópunarmerki, `(!, !=)`.

Ósanngildi í Nanomorpho eru slíkt hin sömu og í Morpho, þ.e.a.s. `false, null`. Annað er skilgreint sem sanngilt.

4.3.9 Kallsegð

Kallsegðir í NanoMorpho eru á forminu `fun(a1, ..., an)`, þar sem $a_1 \dots a_n$ eru 0 eða n breytur sem eru sendar sem inntak inn í fallið `fun`.

4.3.10 Tvíundaraðgerðir

Tvíundaraðgerðir í NanoMorpho er haldið til hagana eins og í Morpho og eru í raun kall á samsvarandi Morpho fall. Tvíundaraðgerðirnar eru eftirfarandi:

`'+', '-', '*', '/', '%', '++'`

4.3.11 Einundaraðgerðir

Einundaraðgerðir í NanoMorpho eru einungis tvær en þær eru:

'-', not eða !

Einundarvirkinn '-' er kall á samsvarandi virkja í Morpho.

4.3.12 if-segð

If-segð í NanoMorpho er lýst með málriti í kafla 3.2.4. Segðin er á forminu:

```
if { (expr1) body1 } elsif(expr2) { body2 } ...elsif(exprn-1) { bodyn-1 } else  
{ bodyn };
```

Þar sem að n getur verið hvaða heiltala sem er ≥ 1 .

Ef að expr_i er sanngilt þá keyrist body_i og farið út úr if segðinni. Ef hún er hinsvegar ósönn, þá er athugað skilyrði fyrir expr_{i+1} og svoleiðis haldið áfram þar til er komið að else klausinni og þá er body keyrt.

4.3.13 while-segð

While-segð í NanoMorpho er lýst með málriti í kafla 3.2. Segðin er á forminu:

```
while{ (expr) body };
```

Ef að expr er ósönn að þá er body ekki keyrt og farið samstundis úr while-lykkjunni. Ef hún er hinsvegar sönn þá er body keyrt og að þeirri keyrslu lokinni er athugað aftur hvort að expr segðin sé sönn, ef hún er það ennþá er body aftur keyrt o.s.frv. þar til segðin reynist ósönn.

4.4 Föll og forrit

Föll í NanoMorpho eru skilgreind í kafla 3.2.2 í málriti.

Að sama skapi eru forrit skilgreind í kafla 3.2.1 í málriti

Athugið við gerð á falli að fallið þarf ekki að taka inn gildi og getur því verið tómt, en mikilvægt er að fallið útbúi gildi við upphafi fallsins þegar það er skrifað. Svo að lokum er kallað á sjálft forritið í `main()` hluta forritsins.

Sýnidæmi um forrit `test.s` má finna á næstu blaðsíðu.

```
// test comment
test()
{
    var a,b;
    a = 2+3;
    b = "HelloWorld";
    if(a == 5) {
        writeln(a);
        writeln(b);
    };

    if(a != 5) {
        writeln("if a is != 5");
    }
    elsif (a < 5) {
        writeln("else if a is < 5");
    }
    else {
        writeln("else a is >= 5");
    };

    return "Return value is " ++ 1;
}
test2(x, y)
{
    var i;
    i = 0;
    while(i != 5){
        i = i + 1;
    };

    if(i == 5) {
        writeln("x equals " ++ x ++ " and y equals " ++ y);
    };
}
main()
{
    var i;
    writeln(test());
    test2(2, 5);
    i = 0;
    while(i <= 3) {
        writeln("while loop " ++ i);
        i = i+1;
    };
}
```
