

Špecifikácia jazyka *Omnomnom*

Omnomnom ignoruje medzery a nové riadky. Príkazy sa vždy končia bodkočiarkou a koniec bloku bodkou, nezáleží na odsadení. Mená premenných a funkcií sa môžu skladať len z písmen a nesmú sa zhodovať s vyhradenými slovami. Platnosť premenných je v rámci bloku, v ktorom boli deklarované. Pokus o prístup k premennej po skončení jej platnosti vyústi v nedefinované správanie. Funkcie sa musia definovať len na začiatku programu.

Vyhradené slová:

- názvy datových typov
- true
- false
- ggwp

Operátory:

- + , - , * , / , % , ^
- = , ++ , --
- == , < , > , <= , >= , /=
- & , | , ~

Dátové typy:

- Základné: `b, i, f, s` (`bool, interger, float, string`)
- Pole `{typ}` (`{b}, {i}, {f}, {s}`)

Deklarácia premenných:

```
typ_nazov;  
typ nazov = hodnota;
```

```
i_x = 5;
b_y = true;
s_z = "omnomnom";
```

Polia (dynamické, jednorozmerné):

```
{typ}_pole = {prvok1, prvok2,..., prvokn};  (pole s n prvkami)
{typ} pole;                                (prázdné pole)
```

pole->g(index);	(vráti prvok na danom indexe)
pole->r(index);	(vymaže prvok na danom indexe)
pole->s(index, prvok);	(nastaví prvok na dané miesto)
pole->a(prvok);	(pridá prvok na koniec pola)
pole->l()	(dĺžka pola)

If:

```
? (vyraz)
    prikaz;
    prikaz;
.
!?(vyraz)
    prikaz;
    prikaz;
.
!
    prikaz;
    prikaz;
.
```

While:

```
@ (vyraz)
    prikaz;
    prikaz;
.
```

For:

```
#(i_it = 0, vyraz, it++)
    prikaz;
    prikaz;
.
```

Funkcie:

```
navratovyTyp_nazovFunkcie(typ1 var1, typ2 var2, ..., typn varn)
    prikaz;
    prikaz;
    ggwp navratovaHodnota;.
```

Externé funkcie:

```
~navratovyTyp_nazovFunkcie(typ1, typ2, ..., typn).
```

Todo:

- tuple [typ1, type2, ..., typen]_tuple = [var1, var2, ... , varn]
- správa pamäte