

DUM č. 19 v sadě

29. Inf-5 RoboLab a Lego Mindstorms

Autor: Hana Křetínská

Datum: 25.06.2014

Ročník: 1AV, 2AV, 3AV, 4AV, 5AV

Anotace DUMu: Výuka robotiky a programování pomocí stavebnice Lego a programu Lego Mindstorms a programování v prostředí BricxCC. Sestavení robotků z Lega a jejich programování na počítači pro žáky 2. stupně ZŠ a první a druhý ročník čtyřletého gymnázia. Proměnné v programování v NXC.

Materiály jsou určeny pro bezplatné používání pro potřeby výuky a vzdělávání na všech typech škol a školských zařízení. Jakékoliv další využití podléhá autorskému zákonu.



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Lego Mindstorms a proměnné v NXC

Typy proměnných

- `int x; // deklarace x`
- `bool y,z; // deklarace y a z`
- `long a=1,b; // deklarace a a b, inicializace a na 1`
- `float f=1.15, g; // deklarace f a g, inicializace f`
- `int data[10]; // pole data deseti nul`
- `bool flags[] = {true, true, false, false}; // množina boolovských hodnot`
- `string msg = "hello world"; // řetězec`

Globální proměnné

- Globální proměnné jsou deklarovány mimo jakýkoliv blok
- Jedna deklarace může být použita ve všech úkolech, funkcích nebo podprogramech
- Její rozsah působnosti začíná v místě deklarace a končí s koncem programu

Lokální proměnné

- Lokální proměnné jsou deklarovány v úkolech a funkcích
- Jsou přístupné jen v rámci kódového bloku, ve kterém jsou deklarovány
- Jejich rozsah platnosti začíná v místě jejich deklarace a končí s koncem bloku, kde jsou deklarovány
- Složený příkaz ohraničený symboly „{“ a „}“ je považována také za blok

Umístění proměnných

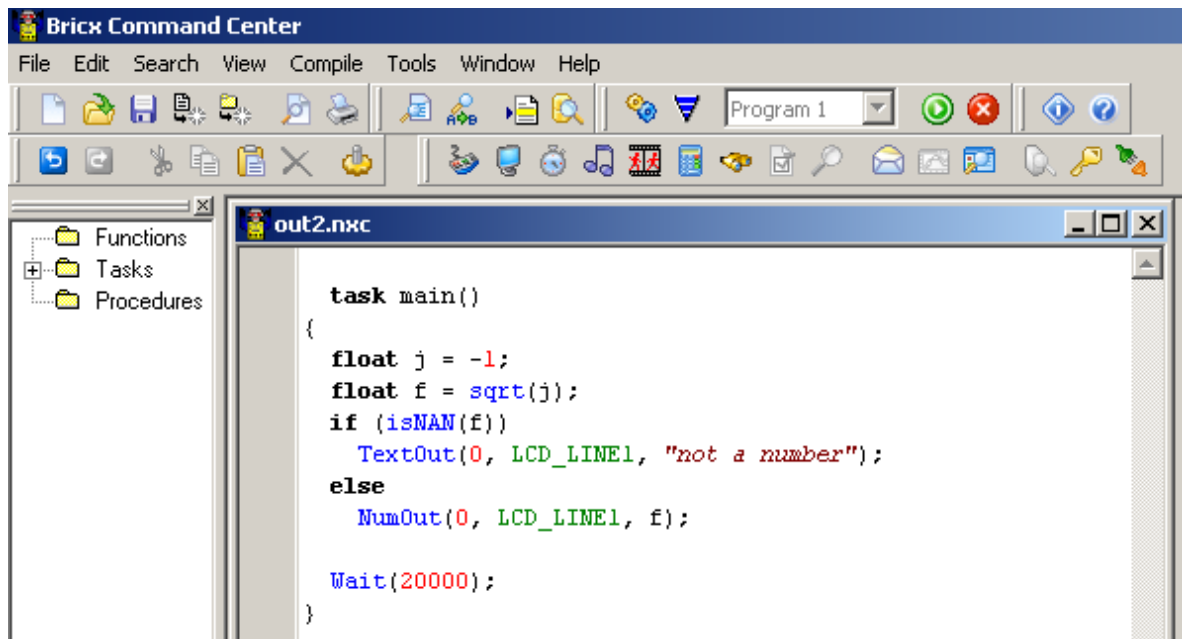
```
int x; // x je globální
task main()
{
    int y; // y je lokální
    x = y; // ok
    { // začátek složeného příkazu
        int z; // deklarace lokálního z
        y = z; // ok
    }
    y = z; // špatně
}
task dva()
{
    x = 1; // ok
    y = 2; // špatně, y není globální
}
```

Výpočet s proměnnými a výstupem na display

Tento program po zjištění, jestli je proměnná j záporná, vypíše na display NXT kostky hlášku, že číslo není v pořádku. Odmocninu ze záporného čísla nemůžeme vypočítat. Když je odmocňované číslo kladné, tak se vypočte a vytiskne na display. Příkazy pro tisk na display:

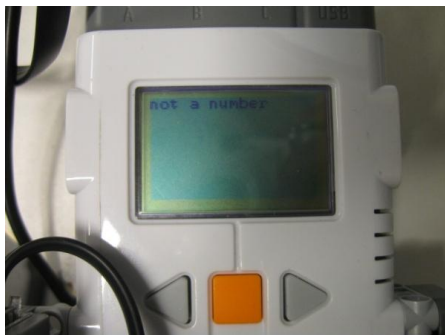
```
char TextOut ( int           x,  
              int           y,  
              string        str,  
              unsigned long options = DRAW_OPT_NORMAL  
            )
```

TextOut(0, LCD_LINE1, "textový řetězec"); tisk řetězce na první řádek
NumOut(0, LCD_LINE2, a); tisk čísla na druhý řádek



```
task main()  
{  
    float j = -1;  
    float f = sqrt(j);  
    if (isNaN(f))  
        TextOut(0, LCD_LINE1, "not a number");  
    else  
        NumOut(0, LCD_LINE1, f);  
  
    Wait(20000);  
}
```

1 Výpočet odmocniny



2 Tisk výsledku na display

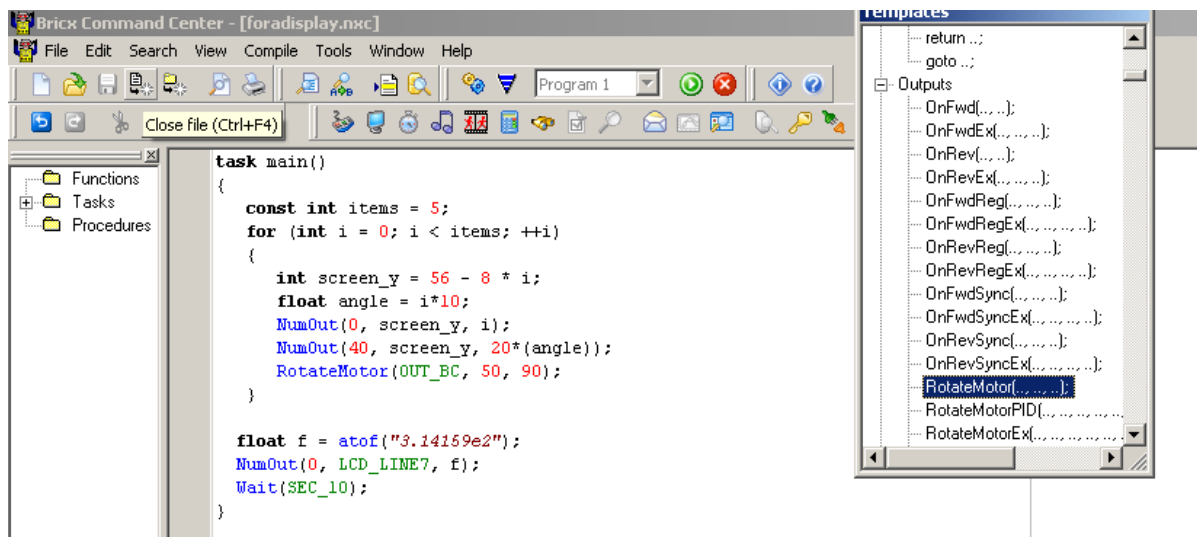
Aritmetické operace v cyklu a tisk na display

Sestavíme program, který vypisuje v cyklu FOR násobky proměnné i a angle po řádcích na display kostky NXT. Při každém průchodu cyklem se motory robotka otočí o 90 stupňů. Po skončení cyklu stanoveném konstantou $items = 5$ se provede převod řetězce "3.14159e2" (číslo v exponenciálním tvaru) na číslo typu float (`atof("3.14159e2")`) s desetinnou čárkou a vypíše se na display.

Příkazy pro otáčení motoru:

```
void RotateMotor      ( byte outputs,  
                        char pwr,  
                        long angle  
                      )
```

```
RotateMotor("ports", "pwr", "angle");
```



3 Tisk hodnot proměnných



4 Hodnoty z cyklu

*Úkol 1: Naprogramujte robotka tak, aby se v cyklu motory otáčely stále o více stupňů.
Například o 20, 40, 60 stupňů atd. až do 360 stupňů.*

Úkol 2: Dále naprogramujte sečtení sta přirozených čísel a výsledek vytiskněte na display.

*Zdroj obrázků: Vlastní tvorba na PC s programem Mindstorms, Bricx Command Center
a vlastní fotografie.*