

Jazyk C

Ladění

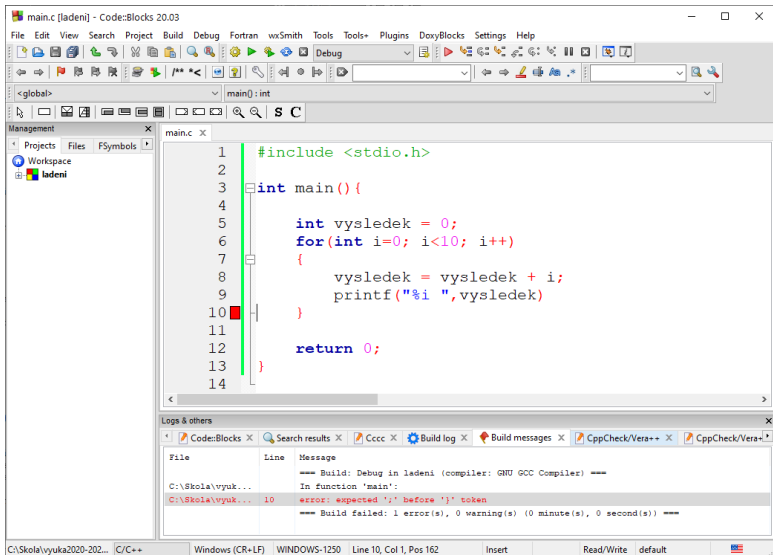
Mgr. Markéta Trnečková, Ph.D.



Palacký University, Olomouc



- **Syntaktické chyby**
- **Sémantické chyby**
 - Nesprávné použití jazyka
 - Chyba algoritmu



The screenshot shows the Code::Blocks IDE interface. The main editor window displays the following C code:

```
1 #include <stdio.h>
2
3 int main() {
4
5     int vysledek = 0;
6     for(int i=0; i<10; i++)
7     {
8         vysledek = vysledek + i;
9         printf("%i ", vysledek)
10    }
11
12    return 0;
13 }
14
```

The IDE's status bar at the bottom indicates the cursor is at Line 10, Col 1, Pos 162. The 'Logs & others' window at the bottom shows the following error message:

File	Line	Message
C:\Skola\vyuk...		==== Build: Debug in ladeni (compiler: GNU GCC Compiler) ====
C:\Skola\vyuk...		In function 'main':
C:\Skola\vyuk...	10	error: expected ';' before '}' token
		==== Build failed: 1 error(s), 0 warning(s) (0 minute(s), 0 second(s)) ====

Example

```
#include <stdio.h>

int main(){
    int a, b, r;
    printf("Zadejte cisla a a b: ");
    scanf("%i %i",&a, &b);
    printf("Vypocet gcd pro %i a %i\n", a, b);
    while(b != 0){
        r = a % b;
        a = b;
        b = r;
        printf("a: %i, b: %i\n", a, b);
    }
    printf("Nejvetsi spolecny delitel je %i", a);
    return 0;
}
```



- Ladění pomocí debuggeru
- Ladící výpisy na `stderr`
- Využití podmíněného překladu
- Self-testy modulů nebo funkcí
- Využití knihovny `assert.h`



```
fprintf(stderr, text, ...)
```

Zavření proudu:

```
fclose(stderr)
```

Přesměrování proudu:

```
fw = freopen("ladeni.log", "w", stderr);
```

Example

```
#ifdef DEBUG /* zacatek podmíneneho prekladu */  
  
/* ladici cast */  
  
#endif /* konec podmíneneho prekladu */
```

Example

```
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>

#ifdef SELF

int funkceA(int a);

int main(int argc, char *argv[]){
    if(argc == 1){
        fprintf(stderr, "Nepredany zadny argument.");
        return 1;
    }
    printf("Vysledek = %d", funkceA(atoi(argv[1])));
    return 0;
}

#endif /* konec SELF */
```


Example

```
#include <stdio.h>
#include <assert.h>

#define VELIKOST 10

int main(){
    int pole[VELIKOST];
    int cislo;
    int index;
    /* Nejaký kod */
    assert(index >=0 && index <= VELIKOST);
    pole[index] = cislo;
    return 0;
}
```

Odstranění assert: definice symbolické konstanty NDEBUG



- 1 Vyzkoušejte si práci s `assert()`.
- 2 Podívejte se na příklady ve skriptech.