

C

zaganjanje programa

```
#include <stdio.h> // importamo standard input output header

int main(){ // ce ne navedemo tipa potem je int
    printf("hello world"); // printf je iz stdio.h
    return 0; // program se je pravilno izvedel zato vrnemo 0
}

helloworld.c -----cc helloworld.c-----> a.out
helloworld.c -----cc helloworld.c -o helloworld-----> helloworld

$.\helloworld
$echo $? // izhodna koda zadnjega izvedenega ukaza
```

```
objdump -d helloworld
```

pri C ne potrebujemo navideznega stroja, ko prevedemo se zapiše datoteka ukazov za točno naš procesor (prenosljivost: OS, tip procesorja)

```
#include <stdio.h>
int f(int n){
    printf("[%d] ", n);
    return n;
}
int main(){
    printf("[%d %d]\n", f(5), f(11));
    return 0;
}
```

na macu dobimo: [5] [11] [5 11]

na win/linuxu: [11] [5] [5 11]

prevajalniki:

- Clang (*cc*)
- GCC (*gcc*)

rezultat je različen ker v C ni določeno (kateri argument) katera funkcija se mora najprej izračunati (vejica ne zagotavlja vrstnega reda - ni sekvenčni operator kot ;)

popravimo:

```
#include <stdio.h>
int f(int n){
    printf("[%d] ", n);
    return n;
}
int main(){
    int first = f(5);
    int second = f(11);
    printf("[%d %d]\n", first, second);
    return 0;
}
```

```
#include <stdio.h>
int main(){
    int first = 10;
    int second = 0;
    printf("[%d]\n", first / second);
    return 0;
}
```

rezultat: [0] zakaj?