

Programiranje 2

Programski jezik C

Kompilacija iz više izvornih datoteka

Inspiracija > <http://poincare.matf.bg.ac.rs/~asamardzic/gnu.pdf>

-polinomi-

Napisati program koji radi sa polinomima:

- a) Definisati strukturu Polinom koja predstavlja polinom (stepena najviše 20). Struktura sadrži stepen i niz koeficijenata. (razmisliti o redosledu navođenja koeficijenata u nizu)
- b) Napisati funkciju koja ispisuje polinom na standardni izlaz u što lepšem obliku.
- c) Napisati funkciju koja učitava polinom sa std. ulaza.
- d) Napisati funkciju za izračunavanje vrednosti polinoma u datoj tački (prednost Hornerovog algoritma).
- e) Napisati funkciju koja sabira dva polinoma.
- f) Napisati funkciju koja množi dva polinoma.

PRIMER BIBLIOTEKE

-polinomi-

- Paralelno dok pišemo funkcije, pišemo i main funkciju koja testira napisane funkcije.
- Želimo da napišemo funkcionalnu biblioteku koju ćemo moći da koristimo u nekim narednim programima. Zato treba izdvojiti deo koji se odnosi na “korisne” funkcije, od main funkcije koja nam je potrebna samo za trenutni program. Od jednog .c fajla, pravimo dva, jedan koji će sadržati main funkciju (main.c), i drugi koji će sadržati sve ostalo (polinom.c). Da bi preveli ceo naš program, treba kompajlirati ova dva fajla. To činimo sledećom naredbom:

gcc -Wall -o polinom polinom.c main.c

- Prevođenje ne prolazi.

PRIMER BIBLIOTEKE

-polinomi-

- Prevođenje nije prošlo jer prevodilac u fajlu main.c nije znao što je tip Polinom, šta je MAX_STEPEN, itd... Ovo rešavamo tako što u main.c prepíšemo typedef iz fajla polinom.c, prepíšemo define direktivu, i stavimo deklaracije funkcija koje smo definisali u polinom.c. Ovo će rešiti prethodne probleme.

Kompilaciju možemo uraditi i na sledeći način:

```
gcc -Wall -c -o polinom.o polinom.c
```

```
gcc -Wall -c -o main.o main.c
```

```
gcc -o polinom polinom.o main.o
```

PRIMER BIBLIOTEKE

-polinomi-

- Prva naredba poziva prevodilac za fajl polinom.c sa opcijama -Wall (štampaj upozorenja prevodioca), -o (fajl koji prevodilac generiše imenuj sa polinom.o) i -c (ne vrši prevođenje do izvršnog fajla, već samo do objektnog). Rezultat ovoga je objektni fajl polinom.o. Taj program je na mašinskom jeziku, ali nije izvršni, jer nije izvršeno linkovanje biblioteka koje su u njemu korišćene, i ostalih objektnih fajlova koji se koriste sa njim.
- Druga naredba radi analogno za main.c.

PRIMER BIBLIOTEKE

-polinomi-

- Ova dva objektna fajla treba ulinkovati međusobno, i ulinkovati sa objektnim kodom standardne biblioteke. To se radi trećom naredbom. Prevodilac gcc prepoznaje da su njegovi argumenti objektni fajlovi i da ne treba da ih prevodi, već samo da ih ulinkuje ispravno.

- Problem koji se sada javio je sledeći:

Ako hoćemo da pravimo neke ispravke u strukturi Polinom, ili da promenimo prototip neke funkcije iz polinom.c, moraćemo da menjamo obe .c datoteke.

Rešenje je da deklaracije iz obe datoteke izdvojimo u zaglavlje (header) polinom.h, i da isti uključimo u obe .c datoteke sa: **#include "polinom.h"**

PRIMER BIBLIOTEKE

-polinomi-

- Možemo primetiti da se zaglavlje koje sami pišemo navode između navodnika za razliku od zaglavlja standardne biblioteke.
- Ovde se podrazumeva da se polinom.h nalazi u tekućem direktorijumu (istom u kome se nalaze i .c fajlovi). Ako se on nalazi negde drugde u fajlsistemu, onda se pod navodnicima mora navesti apsolutna ili relativna putanja od zaglavlja.
- Prevodilac gcc standardne .h fajlove traži na nekim lokacijama koje su unapred određene. U našem slučaju mi moramo da mu specificiramo gde se nalazi naše zaglavlje, na prethodno opisan način, ili nekim opcijama prevodioca.

-polinomi-

- Kako su za prevođenje našeg programa potrebne 3 komande, da bi se ovaj proces kompilacije automatizovao, postoji make alat. Mi specificiramo kako dolazimo do željenog izvršnog fajla u datoteci koja se zove makefile, a make alat za nas izvrši tražene komande.
- Detaljnije uputstvo o make alatu, gcc prevodiocu i još nekim korisnim alatima možete naći u sledećoj skripti:

<http://poincare.matf.bg.ac.rs/~jelenagr/P2praktikum/cas02>