

2. PROGRAMIRANJE KORISTEĆI SE Liberty BASICom

Cilj je da se napišu prvi programi, da se isti pokrenu i da se vide rezultati. Želi se pokazati sva jednostavnost i privlačnost ovog posla. Želi se stvoriti osnova koja će nas uvesti u svijet programiranja i koja će nam omogućiti da steknemo osjećaj za ovaj posao. Nakon toga će slijediti nadgradnja potrebna da bi se shvatilo kako treba kreirati i pisati programe.

Želja nam je da počnemo programirati što prije. Ipak, za to je potrebno i neko predznanje. U našem slučaju će se najprije pokazati kako "stvar" sa programiranjem funkcionira, a potom će se dati i pomenuto predznanje.

2.1 Pokretanje Liberty BASICa

Prvi preduslov za programiranje u nekom programskom jeziku jeste da se isti instalira u računaru. Nakon instaliranja, prevodilac je smješten na tvrdom disku. Liberty BASIC (u suštini je to softver), se sastoji od prevodioca (u žargonu programera odomaćio se izraz *kompajler*, iako bi pravilnije bilo reći *kompilator*) i niza drugih programa. Među ovim programima su i potprogrami za izračunavanje trigonometrijskih funkcija, koji se pozivaju u radnu memoriju računara u momentu kada se u programu dođe do naredbe za izračunavanje neke od ovih funkcija, a za određene vrijednosti argumanata.

Koraci za pokretanje aplikacije Liberty BASICa:

dugme <Start> / Programs ili All Programs / Liberty BASIC v.X.XX

Dobije se LB programski editor, s pozdravnim, odnosno uvodnim ekranom koji opisuje sam LB.

Može se vidjeti da svaka linija teksta počinje apostrofom ('). Linije koje u LBU počinju apostrofom su napomene, odnosno tekstualni komentari. Svi programski jezici omogućavaju pisanje napomena, odnosno komentara, iako svi ne koriste apostrof kao znak koji pokazuje da u nastavku tekućeg reda slijedi komentar. Komentari služe korisniku, a kompajler ignoriše sav tekst nadesno od apostrofa i tako ne opterećuje prevedeni program. Umjesto apostrofa, u istu svrhu se može koristiti i službena riječ REM na početku linije s komentarom.

Korištenje i značaj napomena

- napomene pomažu da kreirani programski listing bude razumljiv. Poželjno je da se prije svake iole komplikovanije sekcije programa stavljaju napomene, kojima se opisuje dio programskog koda koji slijedi poslije napomene. Kasnije, kada sam programer, ili neko drugi bude htio da promijeni program, biće mu puno lakše shvatiti isti preko napomena. Stavljanje napomena jedna je od najvažnijih dobrih navika programera;
- u napomenama se mogu staviti i podaci za identifikaciju programera (ime i kontakt adresa), koji su posebno korisni ako aplikaciju razvija tim programera;
- napomena se može postaviti i u istom redu sa kodom, s desne strane od linije s programskim instrukcijama. I u ovom slučaju se prije navođenja napomene mora upotrijebiti apostrof, ili ključna riječ REM.

Savjet

Kao i većina programskih jezika, i LB dozvoljava uvođenje praznih redova (na primjer za odvajanje pojedinih dijelova programa), praznih prostora u istim, kao i uvlačenje redova kod pisanja programskog koda, s ciljem da se dobije pregledniji i čitljiviji program.

2.2 Koraci do izvršnog programa

Da bi se došlo do izvršnog programa, koji će izvršavati naredbe definisane programom, potrebno je realizovati nekoliko koraka koji su tipični za računarsko programiranje. To su sljedeći koraci:

1. pokretanje programa Editor u okviru Liberty BASIC aplikacije. Liberty BASIC (u daljem tekstu i kao skraćenica: LB), kao i većina programskih jezika, sadrži pored prevodioca (kompajlera) i programski editor (najčešće se zove samo editor). Ovaj program je sličan jednostavnom tekst-procesoru, ili programu za obradu teksta. Korisnik unosi i ispravlja svoj izvorni kod (source code) unutar editora;
2. unošenje, odnosno kucanje programa putem tastature i ekrana računara (konzole računara):

```
INPUT "X=" ; X
Y=X*X
PRINT "Y=" ; Y
STOP
END
```

i snimanje ukucanog programa u fajl s imenom koje bira programer (mefajla.bas, prema sljedećem:

Selektovati meni File / Save As... / selektovati mapu / File name: Prvi.bas / <Save>;

3. prevođenje, odnosno kompiliranje (u žargonu programera *kompiliranje*) izvornog koda u mašinski jezik.

Ako program ima sintaksne greške, nazvane bagovi (engl. bugs), iste treba ispraviti u izvornom kodu, koristeći se programom editor, zatim ponovo snimiti program, te ponoviti prevođenje istog;

4. povezivanje, odnosno linkovanje programa;
5. pokretanje programa.

Treba naglasiti da se prevođenje i povezivanje programa u LB aplikaciji odvija u pozadini, a da korisniku biva vidljivo samo ono što se dobije pokretanjem programa.

Za pokretanje programa treba izabrati opciju Run iz menija Run. U istu svrhu možemo istovremeno pritisnuti tastere <Shift>+<F5>, ili na traci alata izabrati dugme Run.

Važno je napomenuti i da se u aplikaciji LBa može kreirati i snimiti i izvršna verzija programa (s ekstenzijom EXE), odnosno aplikacija, tako da se u meniju Run izabere opcija Create Application (isti rezultat se može ostvariti i istovremenim pritiskom na tastere <Shift>+<F6>). Nakon toga se izvršnom programu da od strane programera izabrano ime, te izabere mapa u kojoj će se isti snimiti.

2.3 Učitavanje postojećeg programa

Generalno, izvorni program se mora ili ukucati u editoru putem tastature, ili učitati s diska, a onda prevesti i povezati, da bi se program mogao startati u okviru aplikacije LBa. Kada se izvorni program unese u editor, prevođenje i povezivanje, pa onda i startanje programa se mogu pokrenuti izborom opcije pokretanja programa u meniju LBa. LB brzo prevede i poveže program (rezultat je program u mašinskom jeziku) i počinje s izvršenjem dobivenog mašinskog programa.

Za fajlove s izvornim programima, LB se koristi ekstenzijom .BAS.ili .bas, kao i većina programa koji potiču od BASICA.

U editor LBa učitati će se postojeći program koji je isporučen na cdu zajedno sa LB aplikacijom. Potrebno je uraditi slijedeće korake:

1. selektovati meni File i opciju Open;
2. selektovati fajl **draw1.bas** i kliknuti na dugme OK. Izvorni program iz fajla draw1.bas pojavljuje se u editoru. Sada postoji mogućnost izmjene izvornog programa. Kako se u ovom slučaju radi o programu koji je neko drugi napisao i koji treba da nam posluži kao primjer, isti ćemo ostaviti neizmijenjen;
3. izabrati opciju Run iz menija Run. U istu svrhu možemo istovremeno pritisnuti tastere <Shift>+<F5>, ili na traci alata izabrati dugme Run. LB prevodi, povezuje i pokreće program. Ako postoje greške u programu, LB će označiti linije u kojima se one nalaze. Prevođenje se neće moći realizovati dok se ne isprave greške;
4. da bi zaustavili program, kliknuti na dugme Close prozora za crtanje.

Na ovom mjestu treba se podsjetiti da se u aplikaciji LBa može kreirati i snimiti i izvršna verzija programa (s ekstenzijom EXE), odnosno aplikacija, tako da se u meniju Run izabere opcija Create Application.

2.4 Kreiranje novog programa

Kreirat će se jedan novi jednostavan program. Jedan od ciljeva nam je da se ovlada osnovnim funkcijama editora.

Da bi se kreirao i pokrenuo novi program, potrebno je uraditi slijedeće korake (prethodno je pokrenuta LB aplikacija):

1. klik na meni File / New BASIC Source File, da bi se dobio čist prozor i pripremio editor za novi program;
2. otkucati u editoru slijedeći program:

```
' Filename: First.bas
'
' Printati na ekran unazad brojeve od 10 do 1
FOR I = 10 TO 1 STEP -1
  PRINT I
NEXT I
PRINT "Brojevi odstampani!"
END
```

i snimiti isti pod imenom First.bas prema slijedećem:

Selektovati meni File / Save As... / selektovati mapu / File name: First.bas / <Save>.

Program sa greškom se neće moći pokrenuti. Sve greške unešene tokom ukucavanja programa će se morati ispraviti, jer ih LB tokom prevođenja otkriva i označava linije koje sadrže ove greške;

3. klik na dugme <Run>, ili <Shift>+<F5> za prevođenje i pokretanje programa.

Rezime

Zašto je potreban prevodilac, da bi se mogao pokrenuti program koji se napiše?

Izvorne programe (pisane na primjer u BASICu, Pascalu, FORTRANu, C++) računar ne razumije. Prevodilac je potreban da bi se izvorni program preveo u mašinski jezik, koji računar razumije.