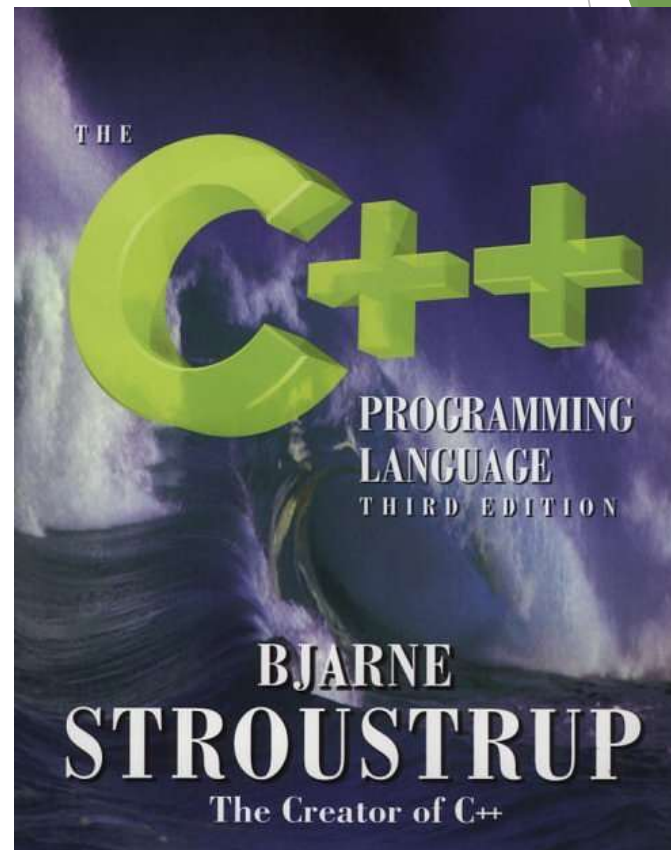


Izrada programa

C++

- ▶ C++ je jezik *opće namjene* za profesionalnu upotrebu.
- ▶ Nastao osamdesetih godina 20.tog stoljeća.
- ▶ Autor jezika - *Bjarne Stroustrup*.

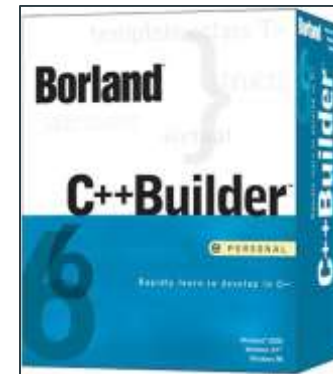


C++ Vrste kompajlera

- ▶ Na tržištu su pristupne razne verzije kompajlera jezika C++ (neke su komercijalne, a neke besplatne).
- ▶ Najpoznatije su:
 - ▶ Microsoft Visual C++, Borland komandni C++ kompajler, GNU djgpp, *Code::Blocs*



djgpp



Izradba programa

► Način izrade programa :

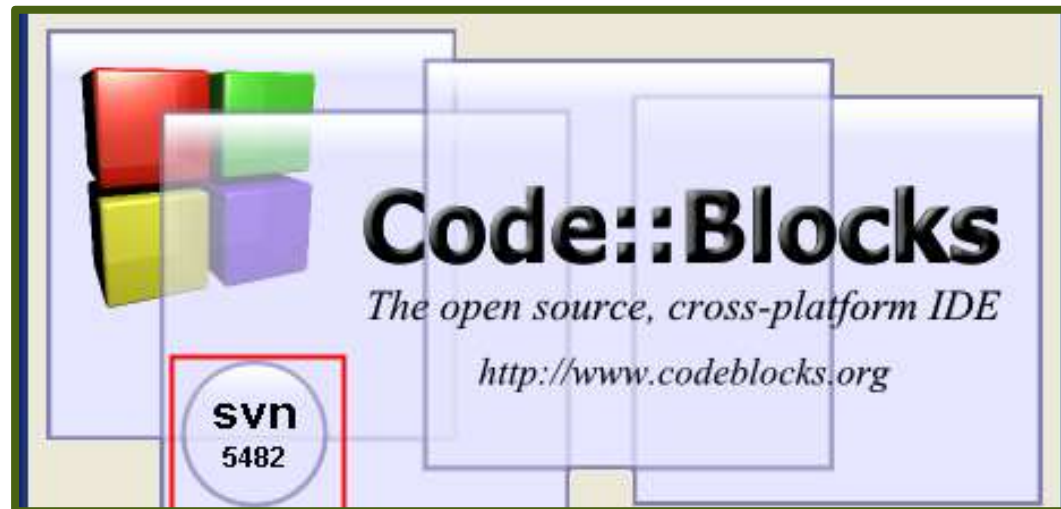
- 1. *pisanje*** izvornog koda,
- 2. *prevođenje*** izvornog koda (kompajliranje),
- 3. *povezivanje*** u izvršni kod,
- 4. *izvršavanje*** i *provjera* programa.

Integrirana razvojna okruženja IDE

- ▶ Za svaki od navedenih koraka koriste se *posebni programi*.
- ▶ Umjesto upotrebe odvojenih programa, koriste se *integrirana razvojna okruženja* (engl. *integrated development environment, IDE*).
- ▶ *IDE* objedinjuju programe za *pisanje* izvornog koda, *kompajliranje*, *povezivanje*, *pohranu*, *izvršenje* i *pronalazak pogrešaka*.

Code :: blocks

- ▶ Osnovni postupci programiranja u jeziku C++ bit će pokazani pomoću integriranog razvojnog okruženja *Code::Blocks*



Datoteka izvornog koda

- ▶ Datoteka izvornog koda (engl. *source code*) je *uputa* za rješavanje zadatka *koji je napisan naredbama programskog jezika*.
- ▶ Izvorni kod moguće je pisati u bilo kojem *programu za uređivanje teksta* (engl. *text editor*).
- ▶ Nastavak **.cpp*



Prevođenje izvornog koda

- ▶ Program prevoditelj prevodi *izvorni kod u objektni kod* (međukorak do strojnog jezika) te *provjerava sintaksu* napisanog izvornog koda.
- ▶ Prevođenjem nastaje *datoteka objektnog koda* (engl. *object code*).



Povezivanje

- ▶ Datoteka objektnog koda *nije izvršni* (engl. *executable*) program i ne može se odmah izvršiti na računaru.
- ▶ U izvršni je oblik pretvara program *povezivač* (engl. *linker*) povezujući *objektnu datoteku s bibliotekama* i drugim potrebnim datotekama.

Biblioteke

- ▶ Biblioteke (engl. *libraries*) su *datoteke* koje *sadrže* gotove, prevedene *funkcije*.
- ▶ Pod pojmom funkcije podrazumijeva se *dio programa* koji obavlja *tačno utvrđeni zadatak*.
- ▶ Funkcije se mogu koristiti u drugim programima tako da ih se uključi iz postojećih biblioteka.

Sintaktičke pogreške

- ▶ Ako ovi programi pronađu sintaktičke pogreške (engl. *syntax error*), ispisuju *poruke* o tom.
- ▶ Otkrivene pogreške treba *ispraviti*, pa ponovo pokrenuti prevođenje.

```
C:\Users\Sanda... 5      error: 'n' undeclared (first use in this function)
C:\Users\Sanda... 5      error: (Each undeclared identifier is reported only once
C:\Users\Sanda... 5      error: for each function it appears in.)
C:\Users\Sanda... 12     warning: implicit declaration of function 'print'
C:\Users\Sanda... 18     error: expected ';' before '}' token
=== Build finished: 4 errors, 1 warnings ===
```

Upozorenja

- ▶ Upozorenja (engl. *warnings*) *ne sprečavaju* prevođenje, povezivanje i izvršavanje programa već ukazuju na dijelove programa koji u određenim okolnostima mogu prouzročiti pogrešku pa je stoga najbolje ukloniti njihove uzroke.

```
C:\Users\Sanda... 5 error: 'n' undeclared (first use in this function)
C:\Users\Sanda... 5 error: (Each undeclared identifier is reported only once
C:\Users\Sanda... 5 error: for each function it appears in.)
C:\Users\Sanda... 12 warning: implicit declaration of function 'print'
C:\Users\Sanda... 18 error: expected ';' before '}' token
=== Build finished: 4 errors, 1 warnings ===
```

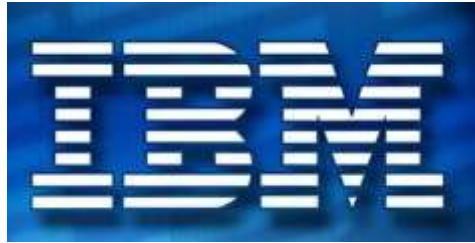
Izvršna datoteka

- ▶ Krajnji rezultat pravljenja programa je izvršna datoteka (*.**exe**)
- ▶ U načelu, izvršnoj datoteci nisu potrebni nikakvi dodaci pa se može izvršavati i bez izvornog programa, prevoditelja, poveziavača, biblioteka itd.



Izvršna datoteka

- ▶ Izvršna datoteka se može odmah izvršiti na računalu za koje je prevedena.



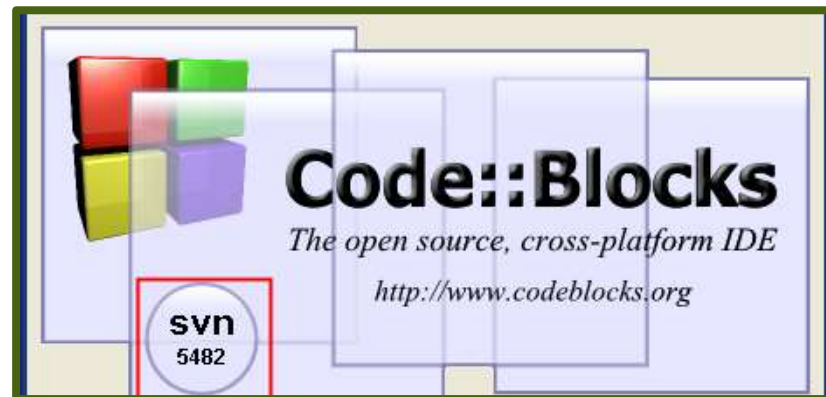
- ▶ Recimo IBM i Apple računari su međusobno nekompatibilni, pa se izvršni program preveden za IBM PC računar ne može izvršiti na Macintosh računaru i obrnuto.

Logičke pogreške

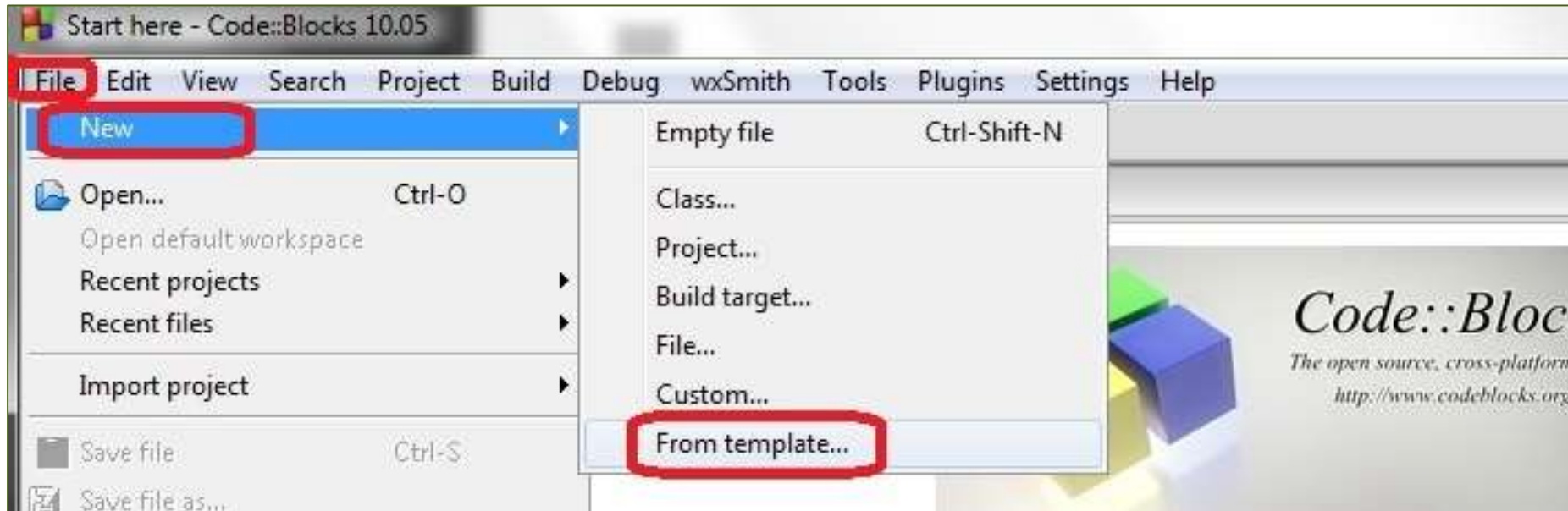
- ▶ Tokom rada, osim sintaksičkih pogrešaka i pogrešaka povezivanja mogu javiti i logičke pogreške.
- ▶ Za otkrivanje logičkih pogrešaka (engl. *run-time error*) potrebno je **provjeriti program** s podacima za koje je unaprijed poznat krajnji rezultat.

Code :: blocks

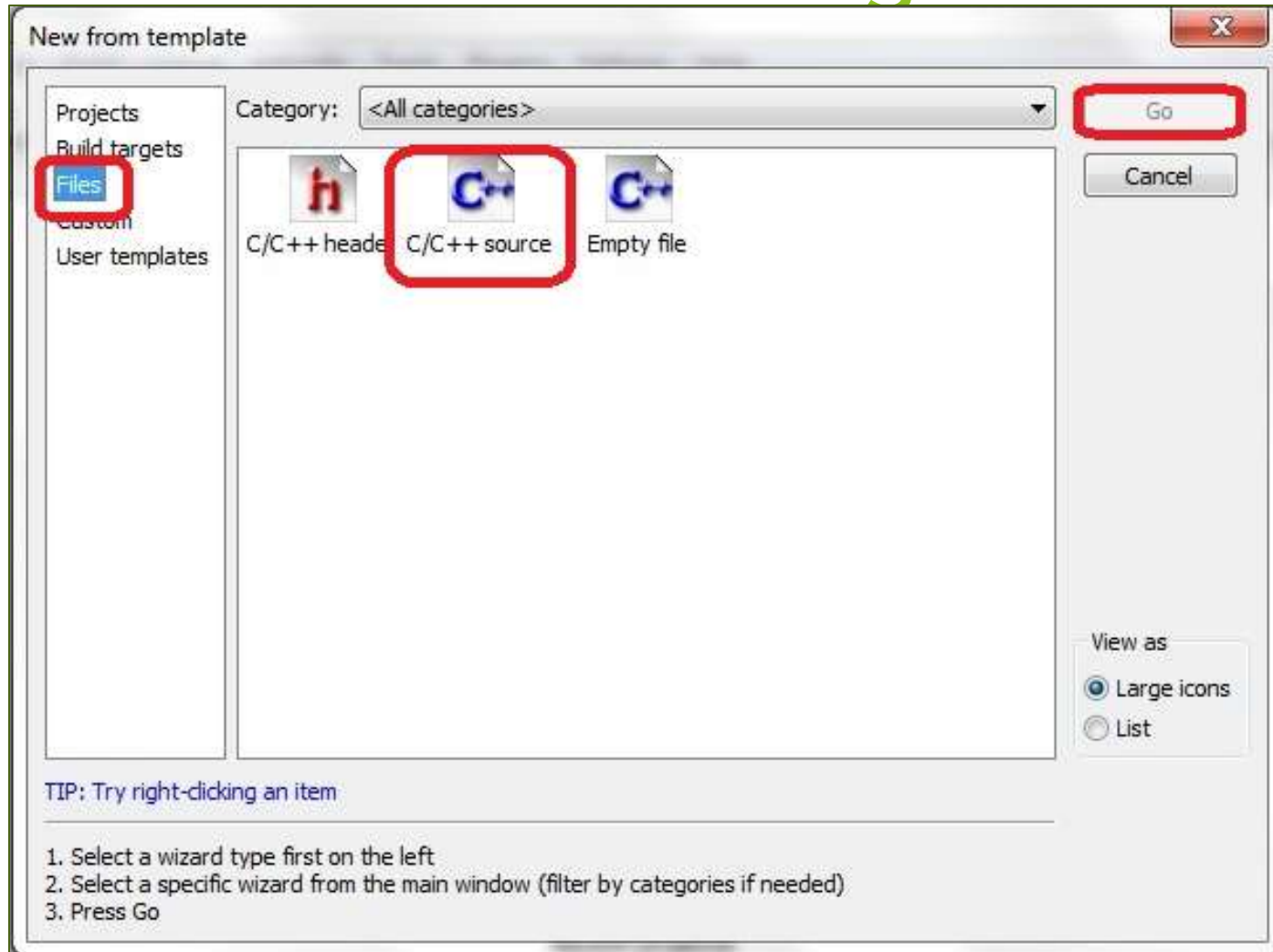
- ▶ Integrirano razvojno okruženja *Code::Blocks* pokreće se na uobičajeni način:
 - ▶ *Start/Svi programi/CodeBlocks.*



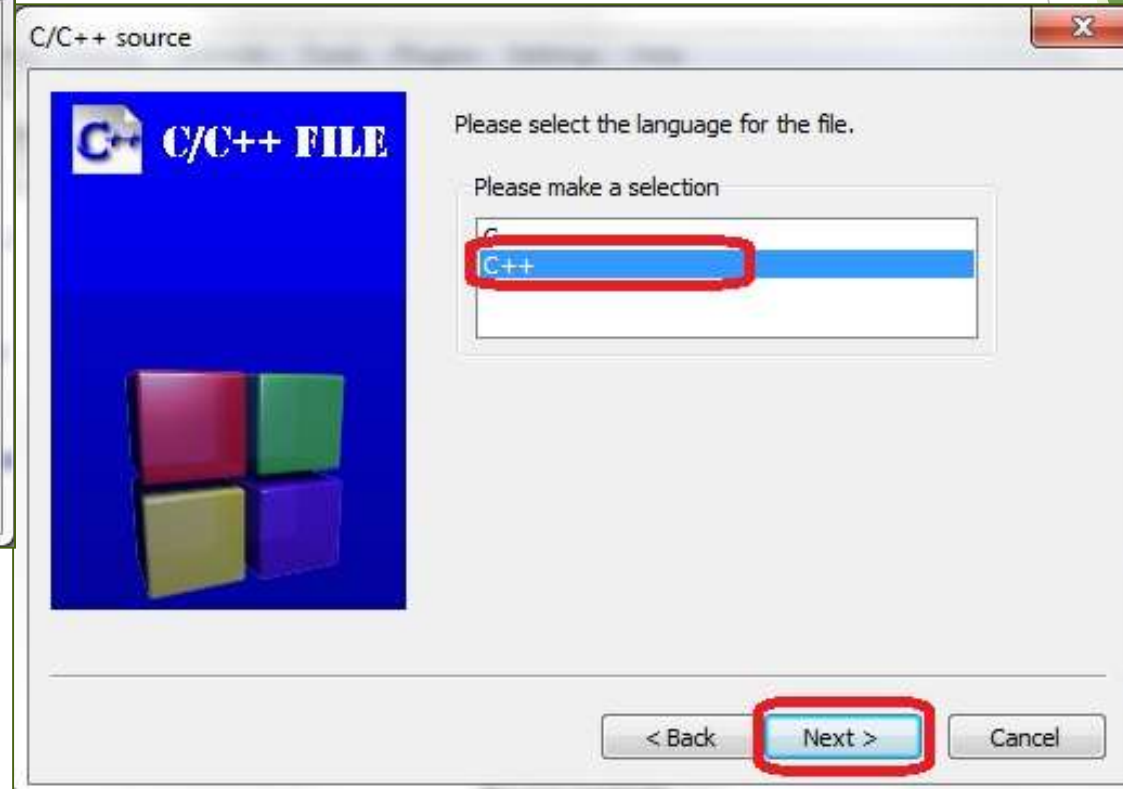
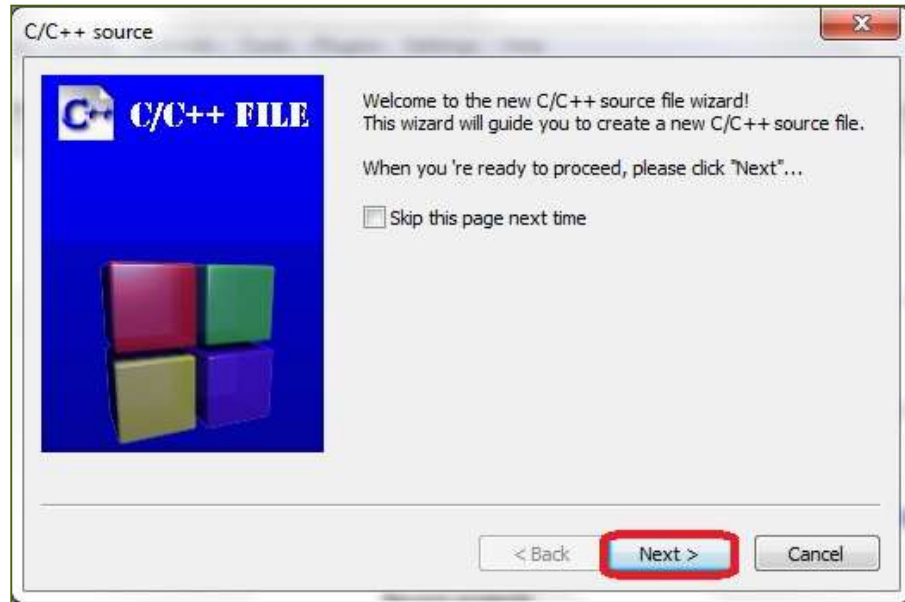
Nova datoteka izvornog koda



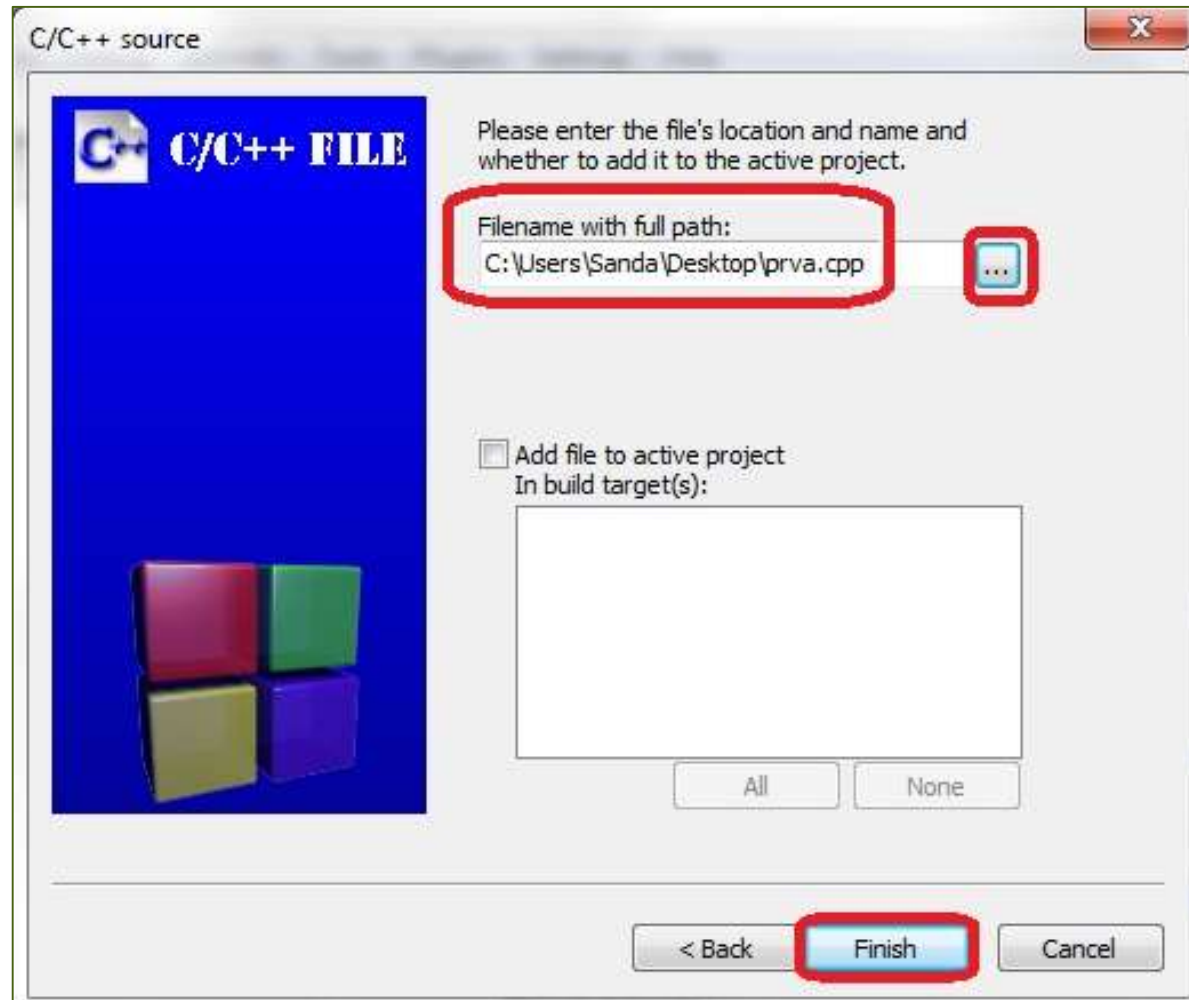
Nova datoteka izvornog koda



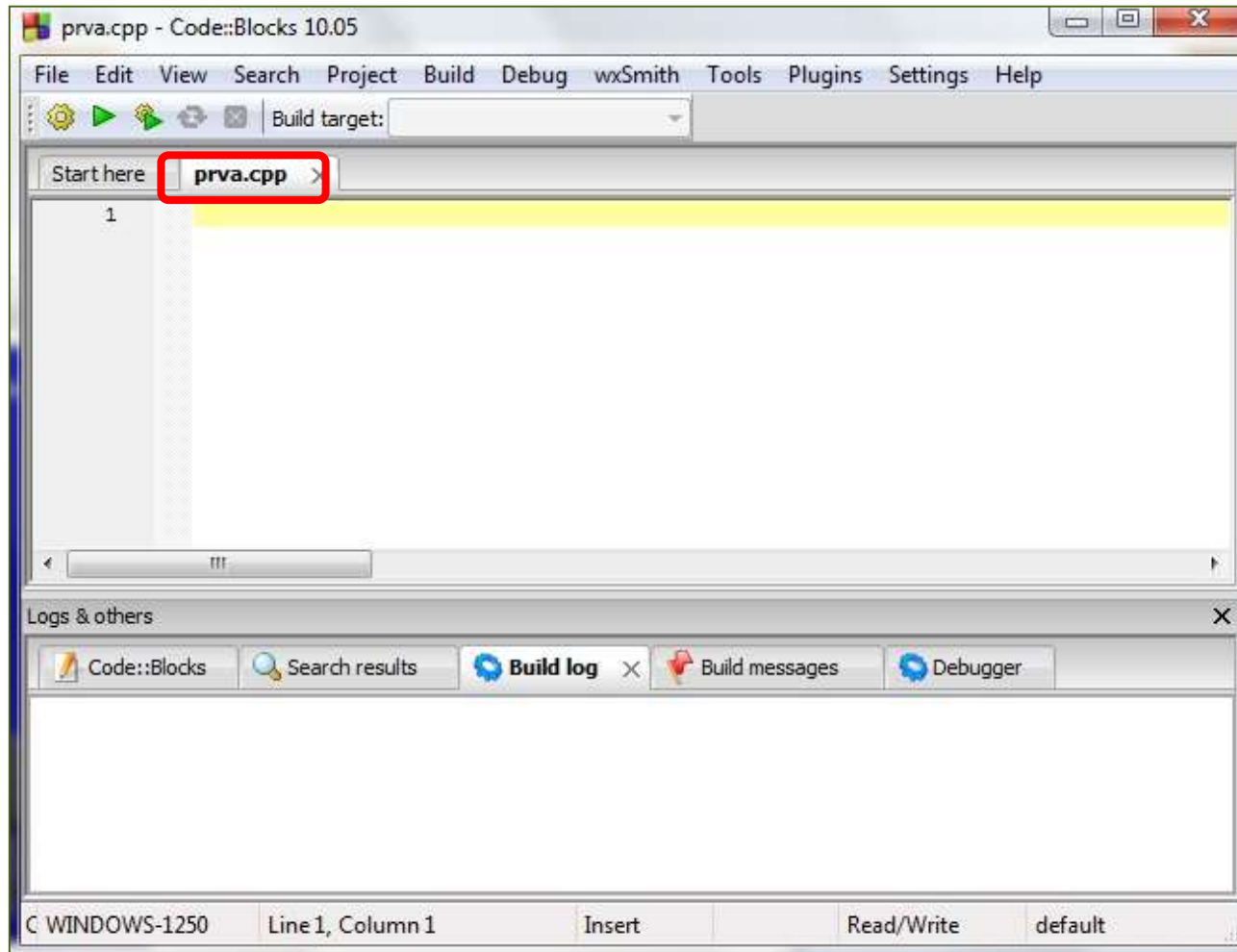
Nova datoteka izvornog koda



Nova datoteka izvornog koda



Nova datoteka izvornog koda



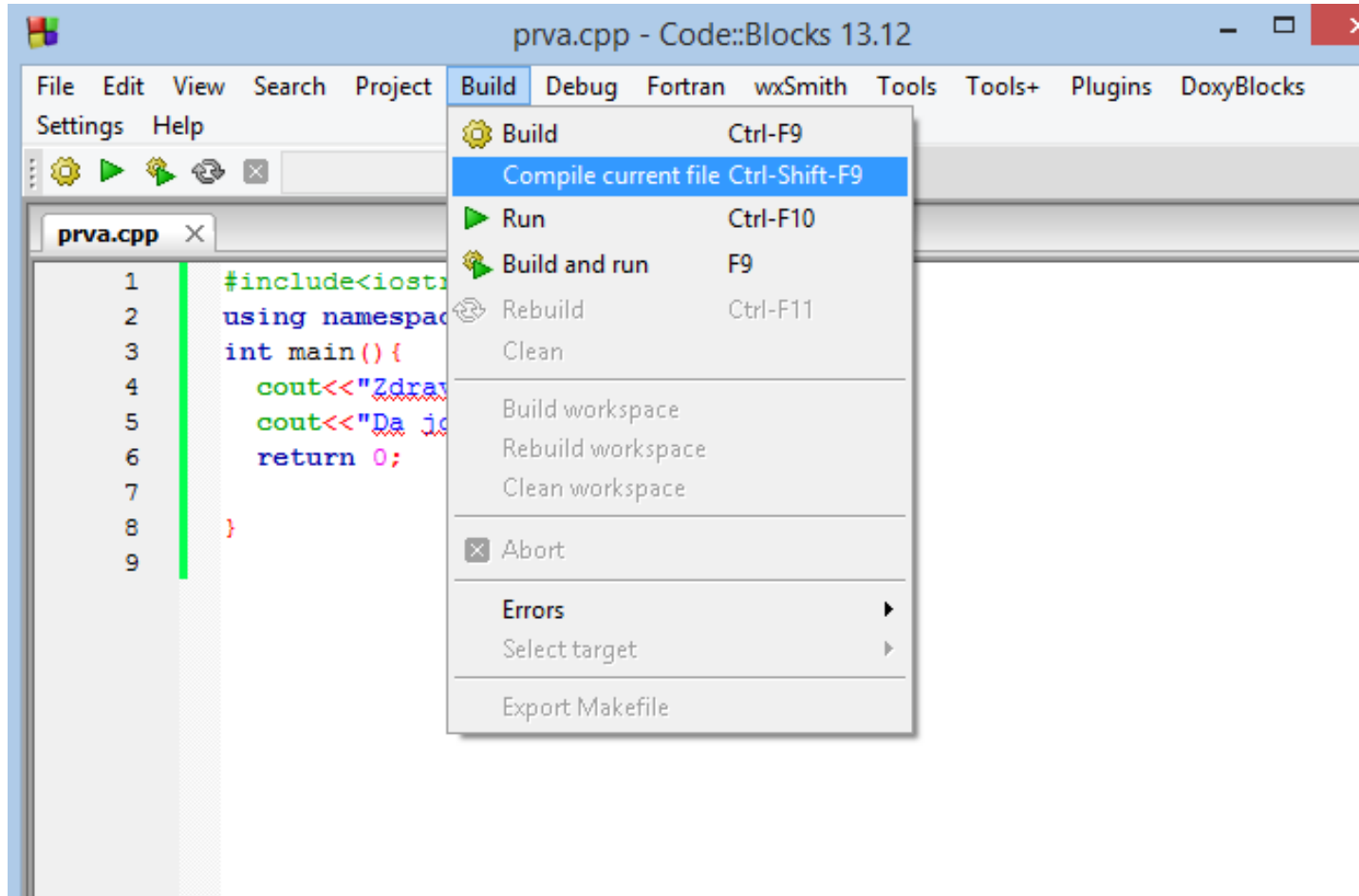
Nova datoteka izvornog koda

- ▶ Datoteku nazvati **prva** pa je spremiti u vlastitu mapu **razred-odjeljenje** prethodno stvorenu na radnoj površini.
- ▶ Unijeti sljedeći kod:

```
1      #include<iostream>
2      using namespace std;
3      int main()
4      {
5          cout<<"Puno uspjeha u ucenju!"<<endl;
6          return 0;
7      }
8
```

```
#include<iostream>
using namespace std;
int main()
{
    cout<<„Zdravo svijete!“<<endl;
    cout<<„Da još malo pišemo!“<<endl;
    return 0;
}
```

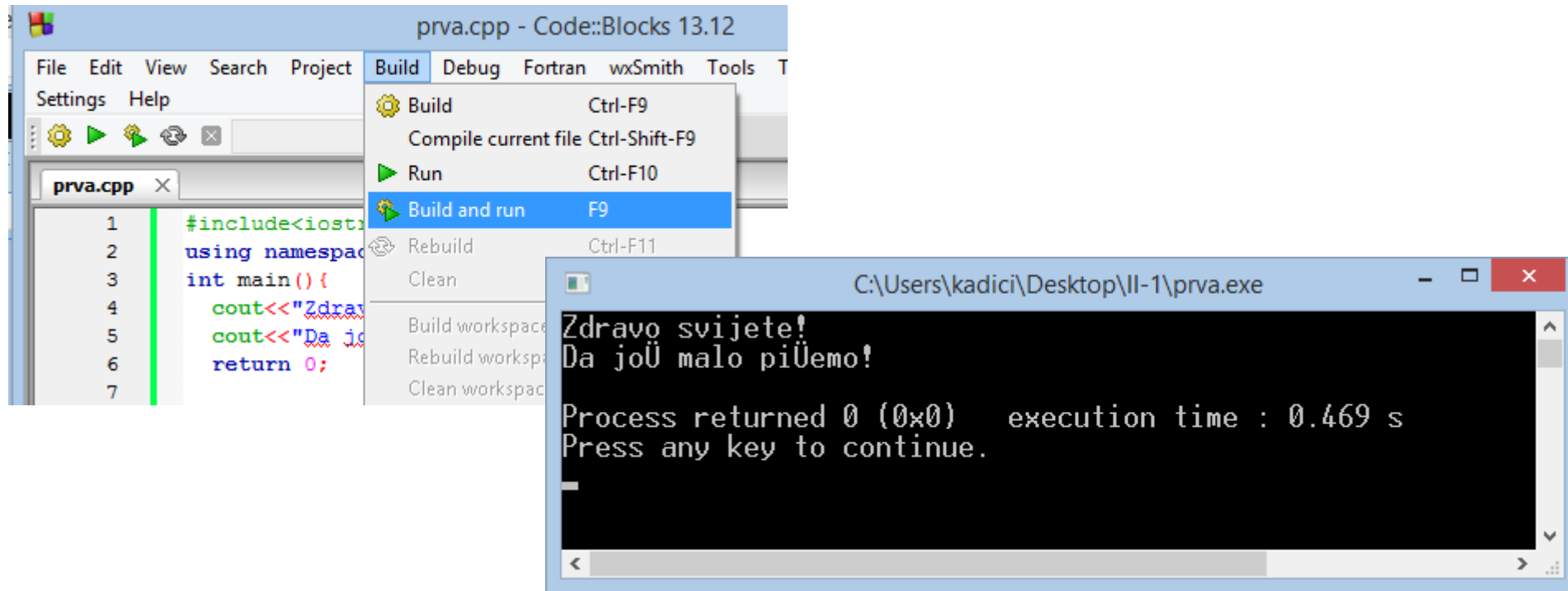
Prevođenje (kompajliranje)



Poruke o pogreškama

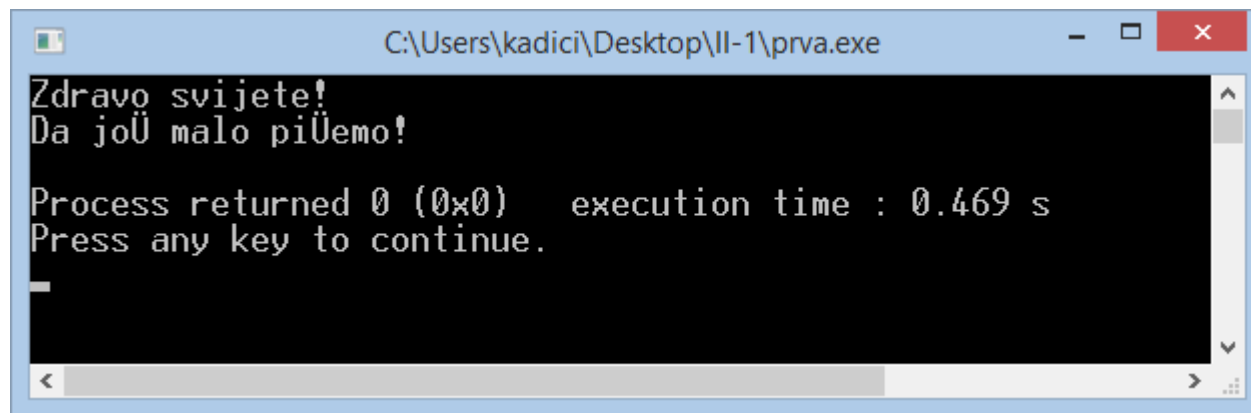
```
Checking for existence: C:\Users\kadici\Desktop\II-1\prva.exe  
Executing: C:\Program Files (x86)\CodeBlocks\cb_console_runner.exe "C:\Users\kadici  
\Desktop\II-1\prva.exe" (in C:\Users\kadici\Desktop\II-1)  
Process terminated with status 0 (0 minute(s), 5 second(s))
```

Povezivanje i izvršenje



Pokretanje izvršne datoteke

- ▶ Pokretanjem izvršne datoteke otvara se *DOS prozor* (ili Command Prompt).
- ▶ U njemu se po potrebi *upisuju ulazni podaci* i *ispisuje rezultat* programa.



The screenshot shows a Windows Command Prompt window titled "C:\Users\kadici\Desktop\II-1\prva.exe". The window contains the following text:

```
Zdravo svijete!  
Da još malo pišemo!  
  
Process returned 0 (0x0)   execution time : 0.469 s  
Press any key to continue.  
-
```

Pokretanje izvršne datoteke

- ▶ Izvršnu se datoteku može pokrenuti dvoklikom s mjesta na kome je pohranjena.
- ▶ Može se dogoditi da se *DOS prozor* u kome je ispisan rezultat *zatvori prebrzo*, pa korisnik ne može pročitati rezultat!

Prebrzo zatvaranje prozora

- ▶ Prebrzo zatvaranje DOS prozora može se spriječiti *dopunom* datoteke izvornog koda:

```
*prva.cpp x
1  #include<iostream>
2  #include<conio.h>
3  using namespace std;
4  int main(){
5      cout<<"Zdravo svijete!"<<endl;
6      cout<<"Da još malo pišemo!"<<endl;
7      getch();|
8      return 0;
9
10 }
11
```

- ▶ Nakon dopune *program čeka unos znaka s tastature* prije nastavka izvođenja.