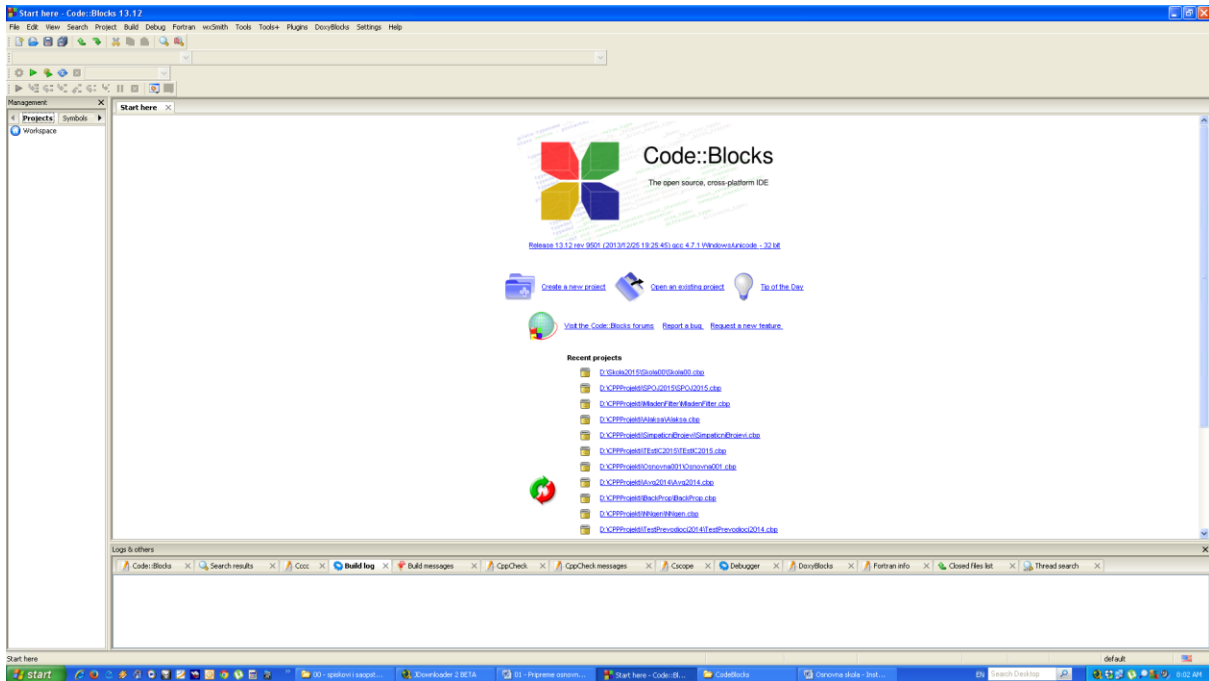
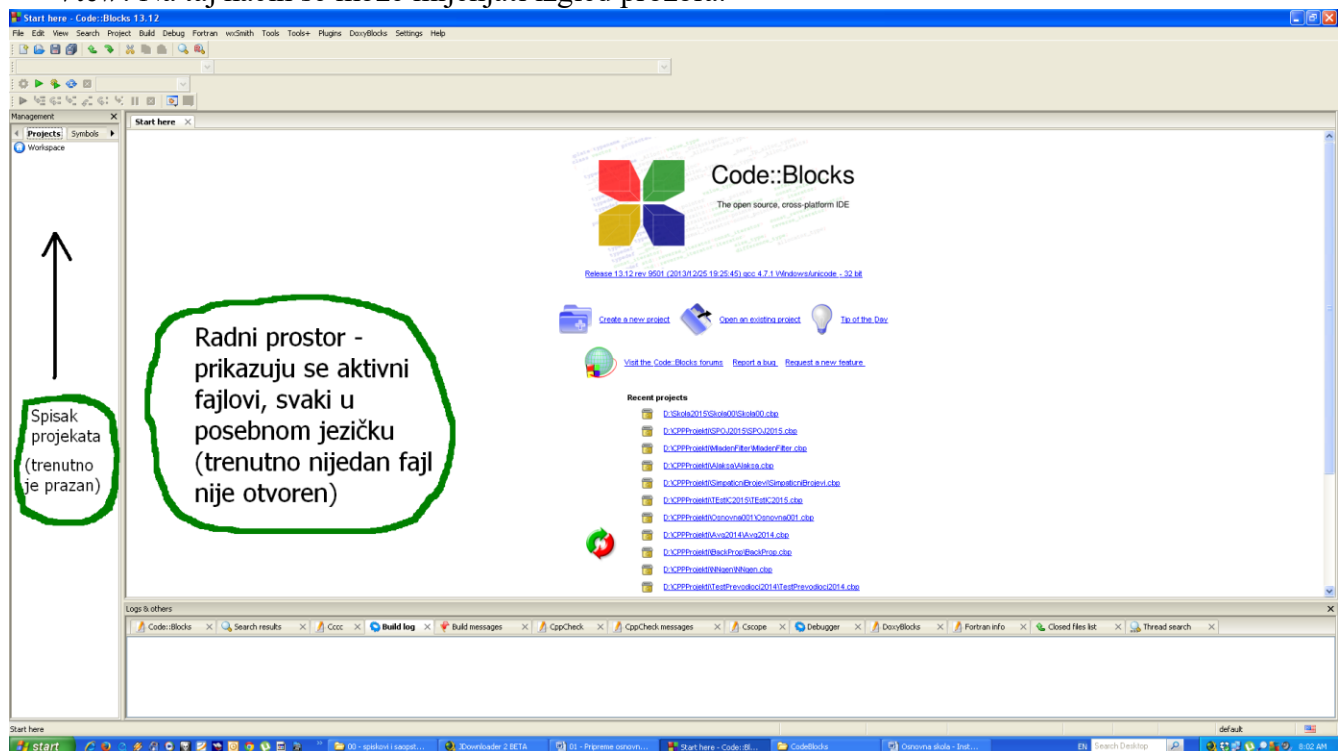


Uvod 13.11.2021. – Početna grupa (osnovna škola)

1. Pokretanje CodeBlocks – dvostruki klik na ikonu na desktopu ili klik na dugme Start/All programs/CodeBlocks ili otvorite folder C:\Program Files\CodeBlocks i dvostruki klik na CodeBlocks.exe.
 - a. Po pokretanju alata otvara se prozor sa slike ispod:

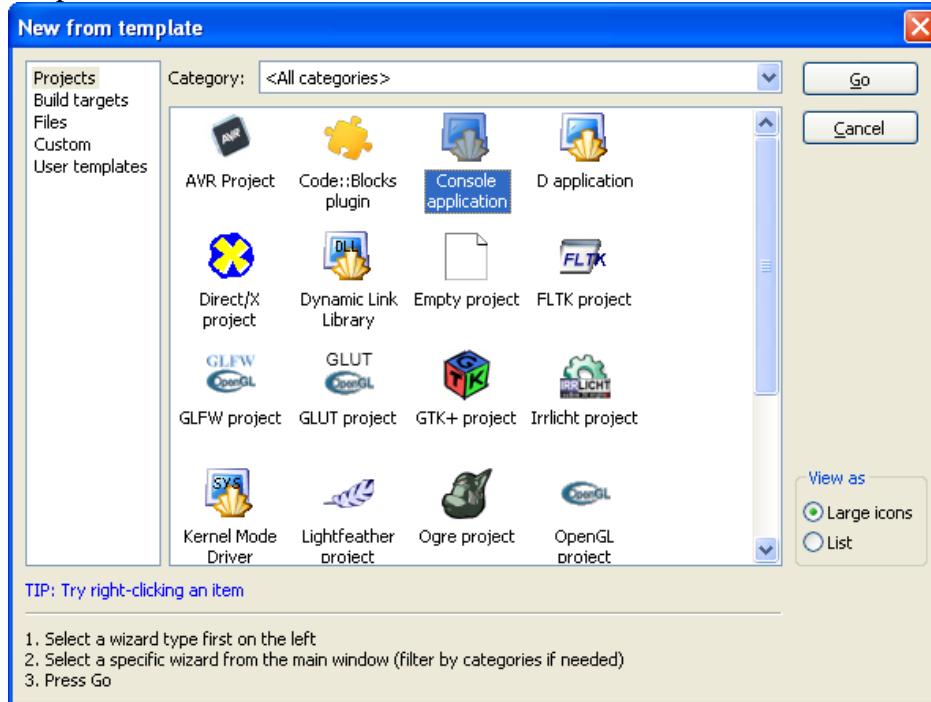


2. Pojedine elemente radnog okruženja možete uklanjati i ponovo prikazivati izborom stavke menija **View**. Na taj način se može mijenjati izgled prozora.

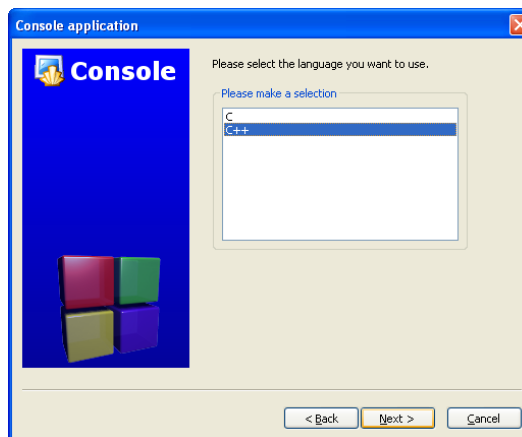


3. Kreiranje prvog projekta.

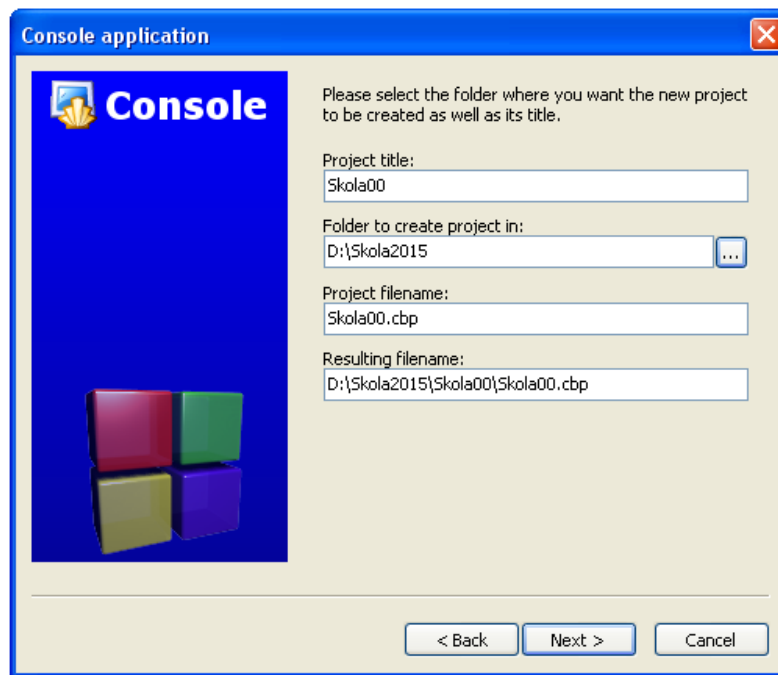
- a. Kreirati folder Skola2015, ako to već niste ranije uradili. Svi učenici koji rade na desktop računarima u sali PMF moraju kreirati folder ili na desktopu računara ili u folderu *My Documents*.
- b. Kliknite na meni **File**, pa podmeni **New...** i kliknite na opciju **Project...**. Otvara se sljedeći prozor:



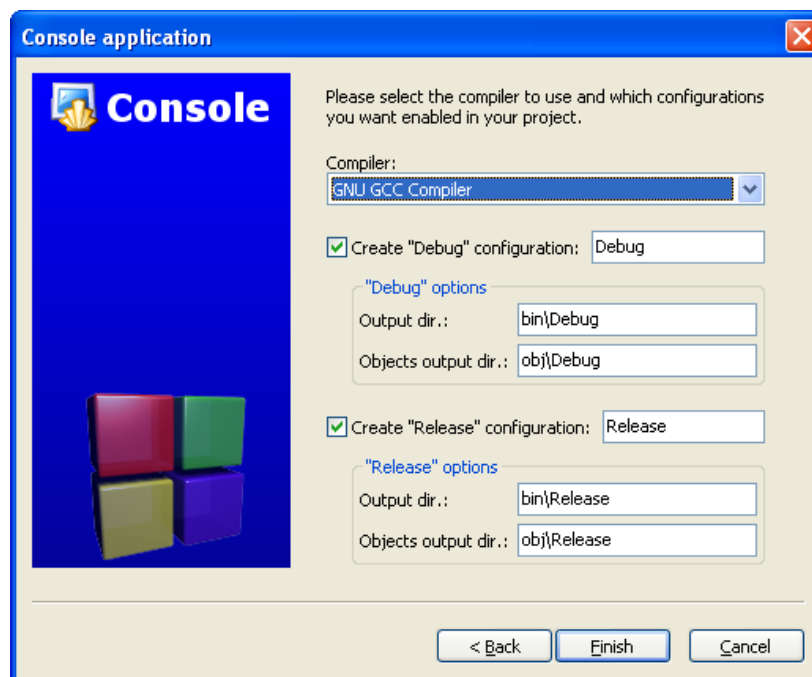
- c. U Skoli programiranja pravićemo samo tzv. konzolne aplikacije (engl. console application) Izaberite opciju **Console application** i kliknite na dugme **Go**. Otvara se prozor za izbor jezika (slika ispod):



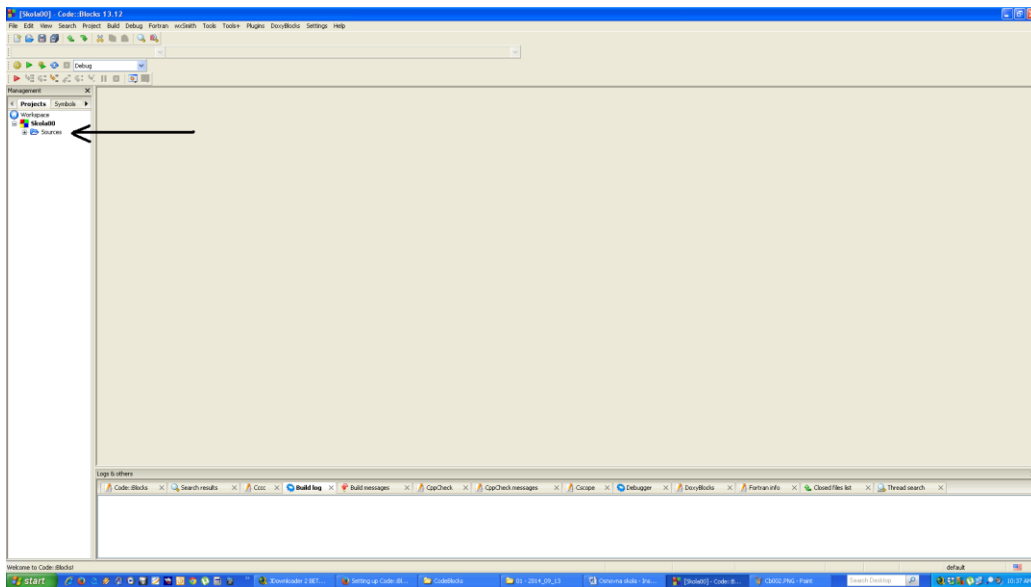
- d. Izaberite C++ iz liste i kliknite na dugme **Next**. Otvara se prozor u kojem morate unijeti ime vašeg projekta i folder u kojem će se sačuvati vaši fajlovi. Za ime projekta (**Project title**) unesite *Skola00*, a u polje **Folder to create project in**, klikom na dugme sa tri tačkice (...), izaberite folder Skola2015 (koji ste ranije kreirali, u koraku 3.a) – vidi sliku ispod.



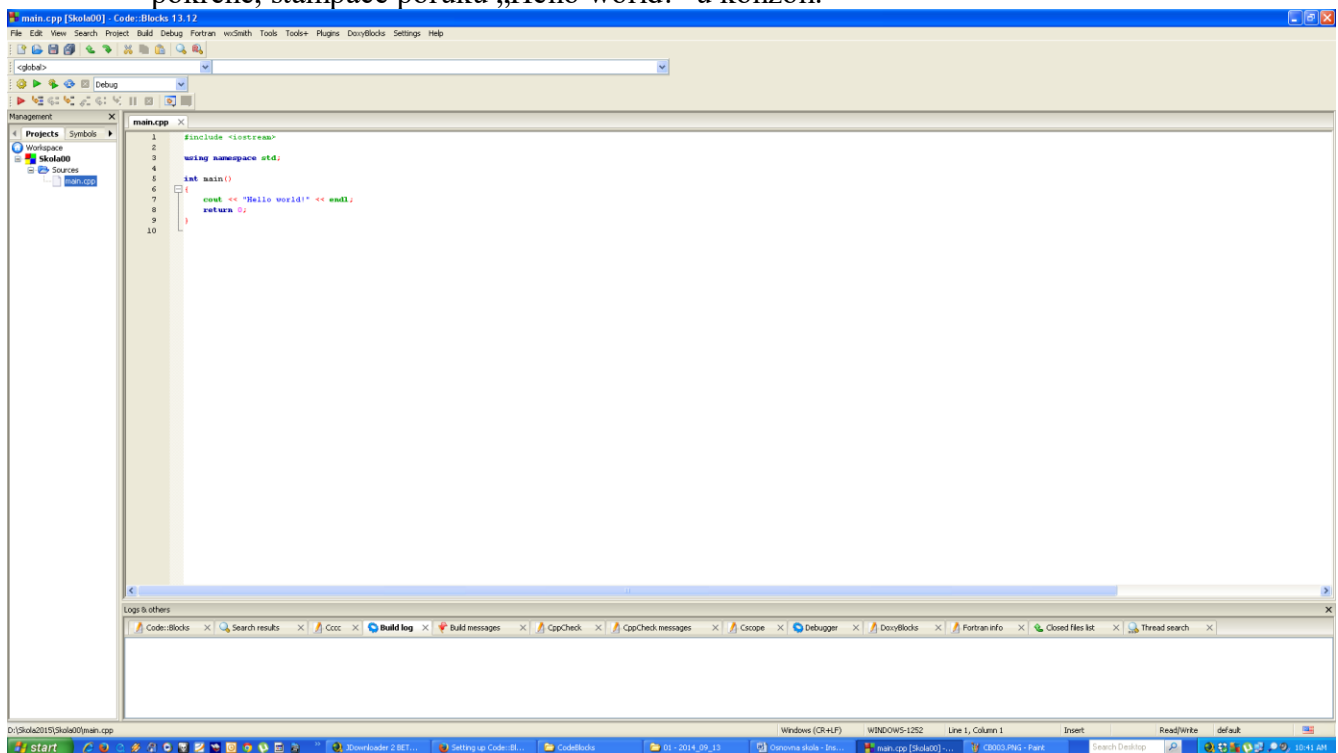
- e. Klikom na dugme *Next*, otvara se prozor za izbor kompajlera (slika ispod). Ne mijenjajte ništa u ovom prozoru i kliknite na dugme *Finish*.




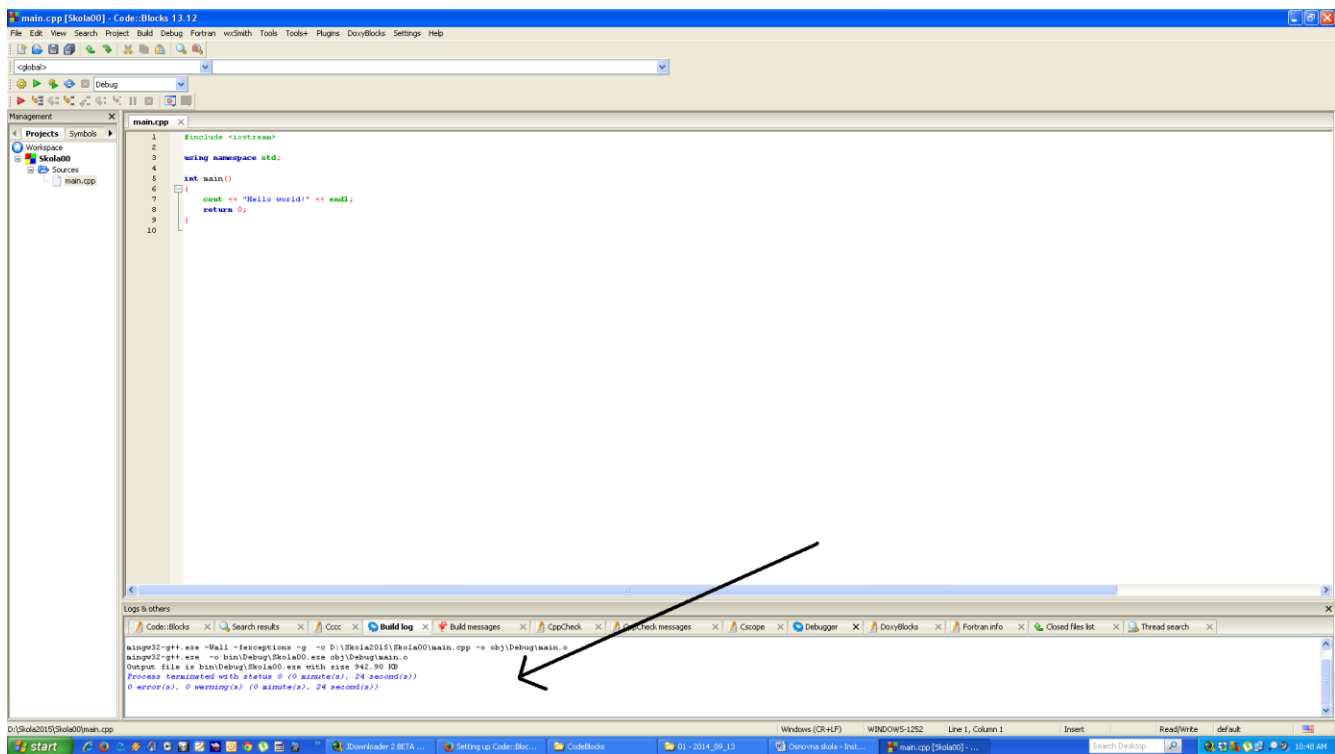
- f. Otvara se prozor sa radnim okruženjem.



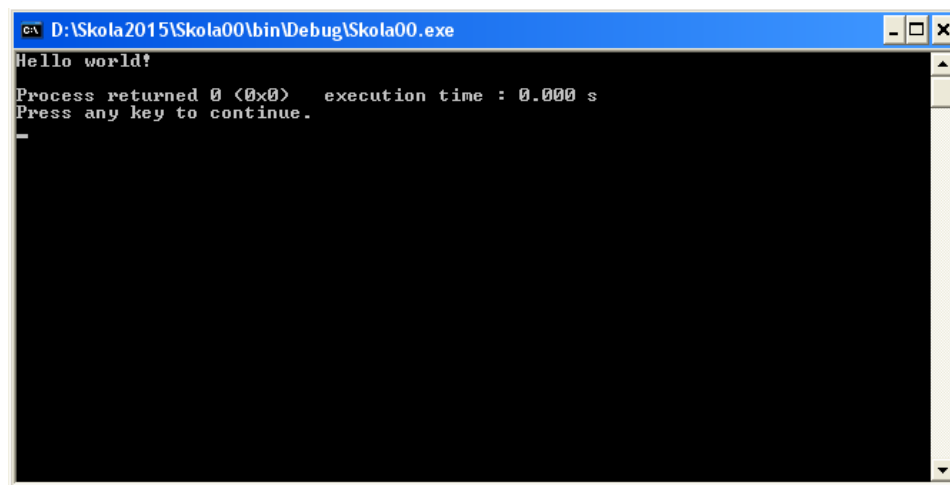
- g. Kliknite na simbol + pored stavke **Source** (označen strelicom na slici iznad) i dvostrukim klikom na fajl *main.cpp* prikazuje se sadržaj tog fajla. CodeBlocks je generisao već neki kod za vas, koji se nalazi u fajlu *main.cpp*. Projekat može da sadrži i druge fajlove, ali u toku Škole programiranja najčešće ćemo koristiti fajl *main.cpp*. Kod iz fajla *main.cpp*, kada se pokrene, štampaće poruku „Hello world!“ u konzoli.




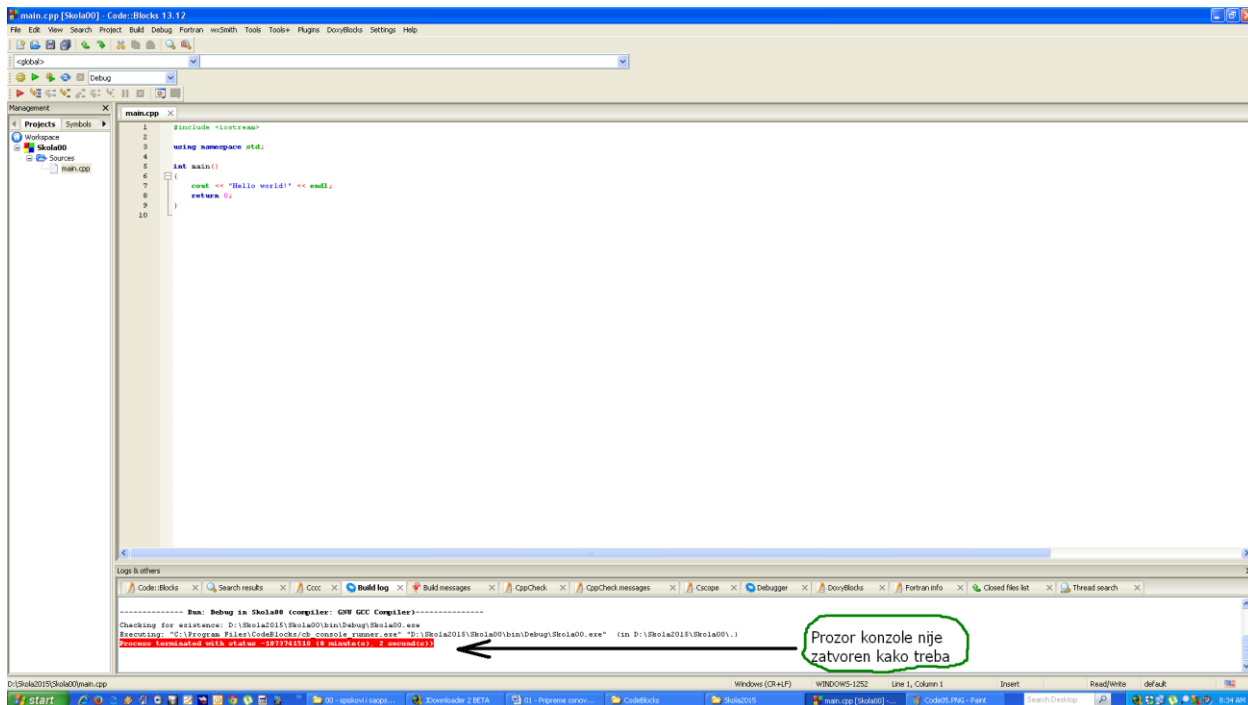
- h. Sada testirajte da li sve radi kako treba. Izaberite stavku menija **Build**, i u njoj prvu opciju koja se takođe zove **Build** (ili kliknite na dugme  na Toolbar-u ili istovremeno pritisnite kombinaciju tastera **CTRL+F9**). Proces prevođenja vašeg programa počinje i može trajati od nekoliko sekundi do nekoliko minuta, zavisno od veličine programa. Sačekajte da u donjem dijelu ekrana dobijete poruku sličnu onoj kao na slici ispod:



- i. Posljednji red poruke vam govori da vaš program nema grešaka i da ga je moguće pokrenuti. Ponovo izaberite stavku menija **Build** i njoj opciju **Run** (ili kliknite na dugme ili istovremeno pritisnite kombinaciju tastera **CTRL+F10**). Vaš program se pokreće u novom prozoru, takozvanoj konzoli, i štampa odgovarajuću poruku (vidi sliku ispod):



- j. Pritisnite bilo koji taster na tastaturi da zatvorite konzolu. To je ispravan način kako se zatvara prozor konzole.
- k. Ako zatvorite prozor konzole klikom na dugme , CodeBlocks vas upozorava da prozor konzole nije zatvoren na ispravan način (vidi sliku ispod):



4. Struktura C++ programa

a. Kod koji se nalazi u fajlu main.cpp (uvećan):

```

main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      cout << "Hello world!" << endl;
8      return 0;
9  }
10

```

- b. Prvi red (red broj 1) sadrži takozvanu direktivu include. Direktive uvijek počinju simbolom # iza koje slijedi neka posebna riječ (u ovom slučaju je to riječ *include*). Kako naš program želi da štampa neki tekst na ekran, potrebno je u program uključiti (otuda riječ *include*) odgovarajuće mehanizme za rad sa ekranom. Svi naši programi će koristiti ekran i tastaturu: tastatura je standardni ulaz (engl. input) a ekran je standardni izlaz (engl. output), Skraćeno se ulaz i izlaz na engleskom jeziku označavaju sa *io*, pa je otuda naziv *iostream*. Uvijek će na početku svakog našeg programa postojati ova direktiva:

```
#include <iostream>
```

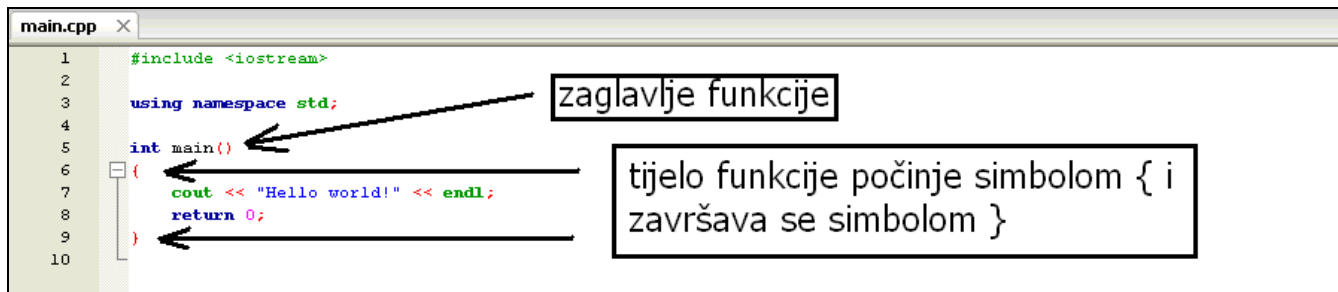
- c. Treći red opisuje takozvani prostor imena (engl. namespace). Svi naši programi će koristiti standardni prostor imena ili skraćeno *std*, pa će poslije direktive *include* uvijek stajati i:

```
using namespace std;
```

- d. Svaki C++ program mora imati mjesto sa kojeg se počinje izvršavati kod. To mjesto je takozvana funkcija *main*. Kasnije ćemo naučiti šta su funkcije i kako da pišemo naše

funkcije. Sam CodeBlocks generiše za nas funkciju `main` i u nju upiše odgovarajući kod. Važno je da posljednja naredba u funkciji `main` bude `return 0`, jer na taj način šalje poruka operativnom sistemu da vaš program uspješno završava rad.

- e. Za sada ćemo sve operacije izvoditi samo unutar funkcije `main` (vidi sliku ispod – sve naše operacije moraju biti između označenih zagrada).



```
main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      cout << "Hello world!" << endl;
8      return 0;
9  }
10
```

zaglavlje funkcije

tijelo funkcije počinje simbolom { i završava se simbolom }

- f. Svi redovi koji se nalaze između simbola `/*` i `*/` su komentar. Komentar može zauzimati više redova (vidi kod ispod). Ako komentar zauzima samo jedan red, onda se taj red može označiti sa dva simbola `/`, tj. `//` (vidi kod ispod).

```
#include <iostream>

using namespace std;
/*
Komentar u više redova.
Obično su to neki netrivialni
podaci o vašem programu.
*/

// Komentar u jednom redu

int main()
{
    cout << "Hello world!" << endl; // komentar
    cout << "Goran Sukovic" << endl; // endl je novi red
    return 0;
}
```

5. Štampanje rezultata: `cout`

- Štampanje rezultata se obavlja preko objekta `cout`. Štampanje se obavlja tako što ono što štampate „šaljete“ ili „prosljeđujete“ na `cout` primjenom operacije `<<` (dva simbola `<`, jedan uz drugi, bez razmaka)
- CodeBlocks vam pruža pomoć pri kucanju teksta. Čim otkucate `cout`, otvara se pomoćna lista koji prikazuje šta je CodeBlocks prepoznao kao ključnu riječ ili funkciju jezika C++. Ako otkucate jedno slovo, lista prikazuje samo one elemente koji počinju tim slovom, itd. Ako otkucate npr. `ca`, lista prikazuje samo one elemente koji počinju sa `ca`. Kada pronađete željeni element, pritisnite `Enter` i CodeBlocks će otkucati taj element umjesto vas. Ovo važi ne samo za već gotove funkcije, već i za funkcije koji vi napišete.
- Ako želite da štampate tekst, tada taj tekst mora biti između dvostrukih navodnika. Takav niz slova ili cifara se zove `string`. `Stringovi` su nizovi karaktera i moraju biti pod dvostrukim

navodnicima. Primjeri stringova su "Hello World!", "Marko Perovic", "a 123 fg", "1245", "Petar\nJasna\nAna", "a", "A", "1"... Više o stringovima učićemo kasnije.


- d. Prelazak u novi red se ostvaruje tako što na cout „pošaljete“ endl ili string "\n". Uočite kako se štampa tekst u sljedećem primjeru.

```
#include <iostream>

using namespace std;
/*
Komentar u vise redova.
Obicno su to neki netrivialni
podaci o vasem programu.
*/

// Komentar u jednom redu

int main()
{
    cout << "Hello world!"; // prvi red
    cout << "Goran Sukovic" << endl; // endl je novi red
    return 0;
}
```

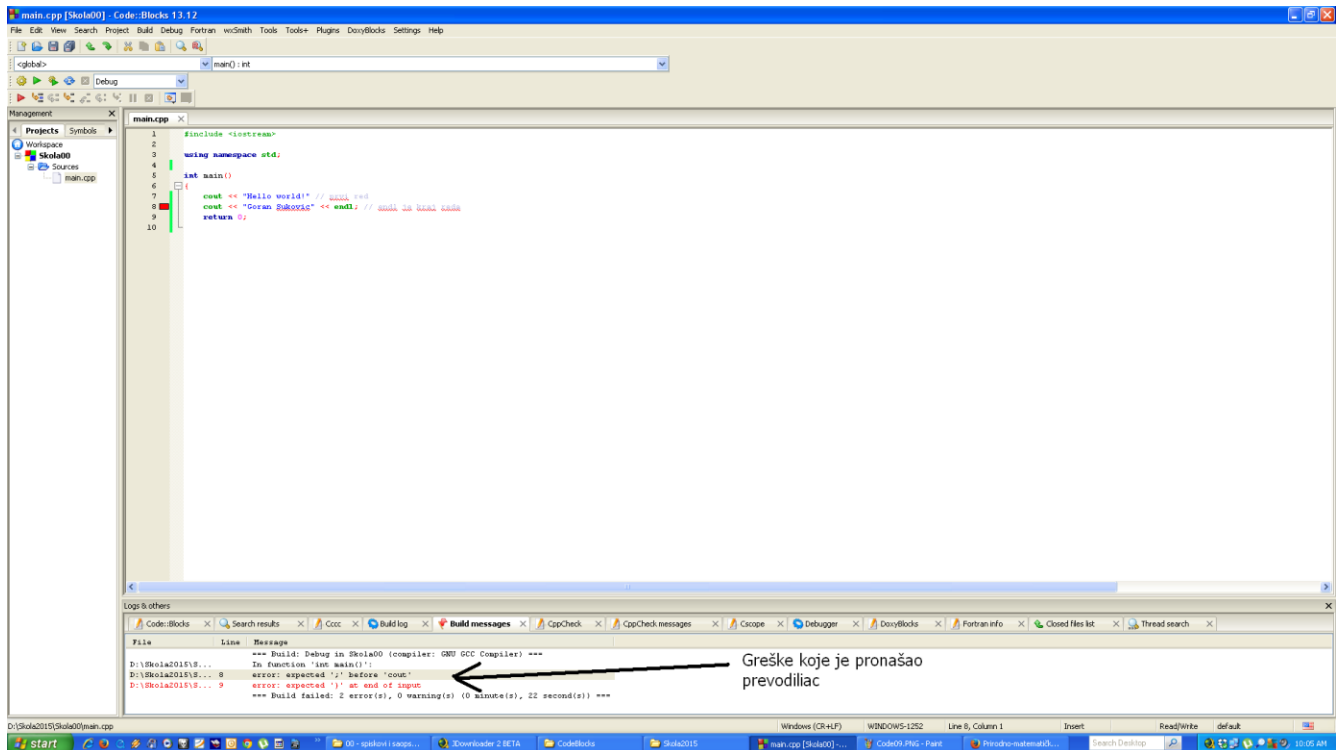
- e. U jednom redu izlaza štampan je tekst "Helo World!Goran Sukovic", bez navodnika. Kako je prva naredba bila štampanje stringa "Helo World!" bez prelaska u novi red, sljedeće štampanje se nastavlja tamo gdje je prethodno završilo, tj. odmah iza simbola "!" .
- f. Ako želimo da štampano više stringova ili brojeva, moramo ih poslati jedan za drugim na cout. Npr, cout<< "The" << "Rolling Stones" <<56 daje "TheRolling Stones56" (obratite pažnju na nedostatak razmaka). Ne zaboravite da poslije svake izmjene koda snimate sadržaj fajla (meni File->Save ili CTRL+S ili dugme  na Toolbar-u) i da ga ponovo prevedete (Build), pa tek onda da ga pokrenete (Run).
- g. Ako vaš program sadrži greške, CodeBlocks ih tokom operacije Build pronalazi i prijavljuje. Pogledajte sljedeći kod koji sadrži dvije greške (ako pažljivo pogledate kod uočićete da nedostaje simbol ";" kod prvog štampanja i da nema zatvorene velike zagrade na kraju koda):

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Hello world!" // prvi red
    cout << "Goran Sukovic" << endl; // endl je novi red
    return 0;
}
```

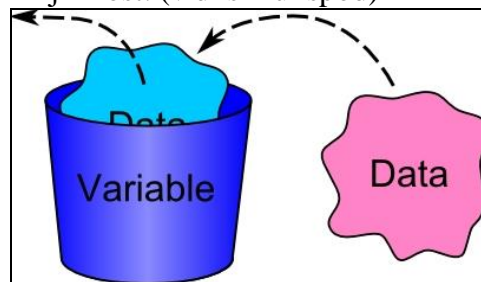
h. Donja slika prikazuje izgled okruženja poslije pokušaja prevođenja neispravnog programa:



i. Dvostrukim klikom na grešku (mjesto označeno strelicom na gornjoj slici), CodeBlocks vas vodi na mjesto u kodu gdje je greška nastala. U ovom slučaju greška je u liniji 8 a opis greške je sljedeći: expected `;` before `cout`. U prevodu, nedostaje simbol `;` prije simbola cout, tj. na kraju sedmog reda. Slično je i za drugu grešku.

6. Dekaracija promjenljivih, tip promjenljive

a. Promjenljiva je ime za neku memorijsku lokaciju. Sadržaj te lokacije, tj. vrijednost promjenljive se mijenja tokom vremena. Dodjeljivanje nove vrijednosti promjenljivoj uništava njenju postojeću vrijednost. (vidi sliku ispod)



b. Kada vam treba promjenljiva, morate zatražiti memoriju koja će čuvati vrijednost te promjenljive. Računaru nije potrebna ista količina memorije za npr. jedan cio broj i jedan razlomljeni (realni) broj. Zbog toga morate navesti kakvog je tipa vrijednost koju želite čuvati u promjenljivoj. Dakle, morate navesti tip i ime promjenljive, i moguće zadati njenu početnu vrijednost. Za cijele brojeve koristi se tip `int` a za razlomljene (realne) brojeve tip `float` ili tip `double`.

c. Vodite računa da je skup cijelih brojeva u računaru konačan, za razliku od matematike, gdje je on beskonačan. Takođe, razlomljeni brojevi se u računaru predstavljaju sa greškom.

Razlomljeni brojevi se pišu sa decimalnom tačkom umjesto decimalne zapete. Npr. 3.54, 2.1245, -12.123. itd. Ne možete zapisati broj $2\frac{3}{4}$ direktno, već samo kao 2.75

d. Primjeri zadavanja (deklaracije) promjenljivih:

```
int i, j, k;
double visina, sirina, m123, ugaoRotacije;
float fl, duzina, obimPravougaonika;
```

e. Ime promjenljive može biti niz slova i cifara koji obavezno počinje slovom. Velika i mala slova se razlikuju, pa su npr. ana12, Ana12 i aN12 različita imena promjenljivih.

7. Naredba dodjeljivanja

- Opšti oblik naredbe je: promjenljiva = izraz
- Prvo se izračuna izraz, pa se njegova vrijednost dodijeli promjenljivoj. Promjenljiva prethodno mora biti deklarirana
- Simboli aritmetičkih operacija: sabiranje (+), oduzimanje (-), množenje (*), dijeljenje (/), modulo – ostatak pri dijeljenju (%)
- Primjer:

```
k = 32;
p = visina * sirina;
k = j/10 + k%10;
a = (i+j+5.62)*(h-1/2) - a/2;
obim = 2*visina + 2*sirina;
```

- Svi operacije se izvršavaju s lijeva u desno, osim operatora dodjeljivanja koji se izvršavaju zdesna ulijevo. Npr. $7+4+8$ se izračunava kao $(7+4)+8$.
- Vodite računa da ako su svi argumenti u izrazu cijeli brojevi, rezultat je također cio broj, čak i ako se koristi dijeljenje. Npr. $13/4$ daje rezultat 3, a ne 3.25. Ako želimo tačan rezultat, moramo pisati $13.0/4$ ili $13/4.0$ ili $13.0/4.0$. Obratite pažnju da je rezultat izraza $13/4*2.0$ jednak 6.0 a ne 6.5.

8. Zadaci urađeni na času. Za svaki zadatak napravljen je poseban projekat i otkucan kod unutar metoda main.

- Kreirati projekat **Skola001**. Napisati program koji učitava dužine stranice pravougaonika i štampa njemu površinu a zatim štampa poruku „Hello World!“ i vaše ime.

```
#include <iostream>

using namespace std;

int main()
{
    float a,b; // stranice pravougaonika
    float površina;
    cin >> a >> b;
    površina = a*b;
    cout << površina << endl;

    return 0;
}
```

9. Zadaci za vježbanje. Za svaki zadatak napraviti poseban projekat (npr. ZadatakA, ZadatakB...), u projektu napraviti klasu *Rjesenje* i otkucati kod unutar funkcije *main*.
- Štampati poruku „Hello World! Vase ime i prezime“ u jednom redu. Npr. moja poruka je:
Hello Word! Goran Sukovic.
 - Štampati dan, mjesec i godinu tvog rođenja, po jedan broj u redu. Npr. moj datum je:
6
12
1966
 - Štampati dan, mjesec i godinu rođenja, u jednom redu, sa po jednim blankom između svaka dva broja.
6 12 1966
 - Učitati dva cijela a i b štampati njihov zbir, razliku, proizvod, količnik i ostatatak pri dijeljenju.
 - Učitati dva razlomljena broja a i b koji predstavljaju dužine stranica pravougaonika i štampati njegov obim i površinu.
 - Učitati cio broj m i odrediti i štampati njegovu cifru stotina.
 - Četvorocifreni prirodan broj m podijelimo na dva dvocifrena broja: prvi broj čine cifra hiljada i cifra desetica (tim redom), a drugi broj čine cifra jedinica i cifra stotina (tim redom). Napisati program koji učitava cio broj m ši tampa zbir dobijenih dvocifrenih brojeva. Npr. ako je $m=1532$, tada se sabiraju 13 i 25 tj. rezultat je 38.
 - Za dati realan broj x izračunati vrijednost izraza $y = 7(x-3)^2 - 6(x-3) + 5$.
 - Za dati realan broj x izračunati vrijednost izraza $y = \frac{7(x-1)^2 - 6(x-2) + 5}{2x + \frac{x^3}{x^2 + 1}}$.
 - Dat je realan broj a. Koristeći pomoćne promjenljive i najviše tri operacije množenja izračunati a^8 . Napomena: za neki broj x, sa x^n označavamo proizvod $\underbrace{x \cdot x \cdot \dots \cdot x}_n$
 - Od početka dana (tj. od ponoći) prošlo je tačno n sekundi, n cio broj. Odrediti i štampati broj sekundi proteklih od posljednjeg punog sata.
 - Dani u sedmici označeni su na sljedeći način: 1 – ponedjeljak, 2 – utorak, ..., 6 – subota, 7 – nedjelja. Dat je cio broj k, $1 \leq k \leq 365$. Koji je dan u sedmici bio k-ti dan u godini, ako prvi januar pada u utorak? Štampati broj k-tog dana. Npr. ako je $k=10$, štampati broj 4, jer je 10. januar četvrtak. Smatrati da godina nije prestupna.