

## MS Visual Studio .NET 2003 - Upoznavanje sa sučeljem

### 2. domaća zadaća

Unutar ove domaće zadaće treba samostalno usvojiti sljedeća znanja:

- kreiranje projekta,
- unos C programa,
- prevođenje programa,
- ispravljanje sintaktičkih pogrešaka,
- pokretanje programa:
  - o bez korištenja načina za ispravljanje pogrešaka
  - o u načinu za ispravljanje pogrešaka
- postavljanje prekidnih točaka,
- pregled vrijednosti varijabli u prekidnim točkama.

Za početak, služeći se tekstom unutar dokumenta "*Upute za rad s VisualStudio .NET 2003*", (koji možete pronaći u repozitoriju *Upute*) kreirajte novi projekt i dodijelite mu neko proizvoljno ime. U daljnjem tekstu pretpostavit ćemo da je naziv projekta Vježba2, pa će se takav naziv pojavljivati na slikama.

Slijedeći upute (str. 4 iz gore navedenog dokumenta) dodajte novu programsku datoteku u vaš projekt i u nju prepisite sljedeći program (koji određuje najveći od tri upisana broja):

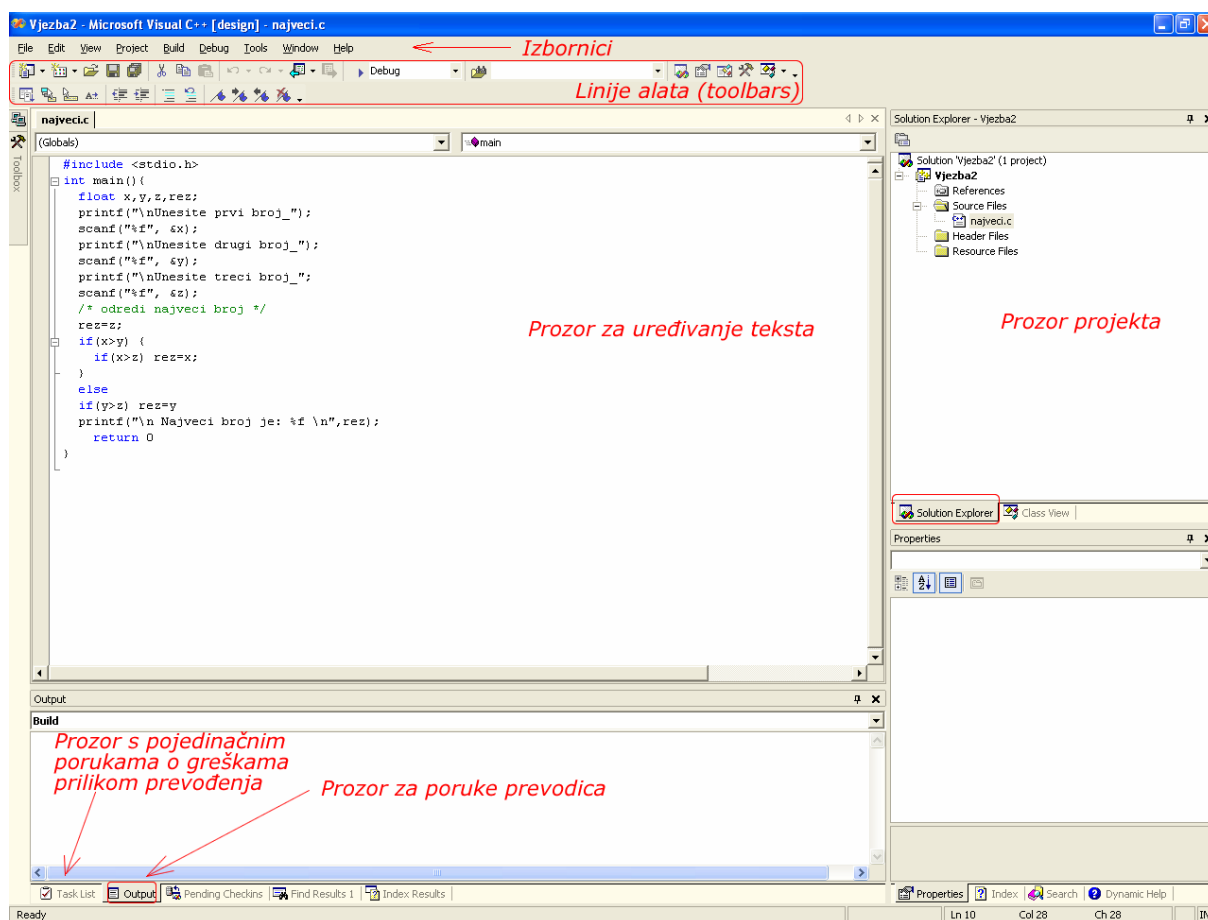
```
#include <stdio.h>
int main(){
    float x,y,z,rez;
    printf("\nUnesite prvi broj_");
    scanf("%f", &x);
    printf("\nUnesite drugi broj_");
    scanf("%f", &y);
    printf("\nUnesite treci broj_");
    scanf("%f", &z);
    /* odredi najveci broj */
    rez=z;
    if(x>y) {
        if(x>z) rez=x;
    }
    else
        if(y>z) rez=y;
    printf("\n Najveci broj je: %f \n",rez);
    return 0
}
```

*Napomena: U primjeru na slikama datoteku smo nazvali najveci.c*

Nakon kreiranja projekta i unosa programskog koda trebali biste imati ekran nalik ekranu na slici 2.1.

Na slici je označeno nekoliko osnovnih elemenata koje Visual Studio.NET 2003 sučelje sadrži. Pomoću izbornika moguć je osnovni pristup svim funkcijama razvojnog okruženja. Ako uz opciju izbornika slijede tri točkice (...), odabirom te opcije otvara se novi izbornik. Preko linije alata možete, pritiskom lijeve tipke miša iznad određene ikone, obaviti neke često

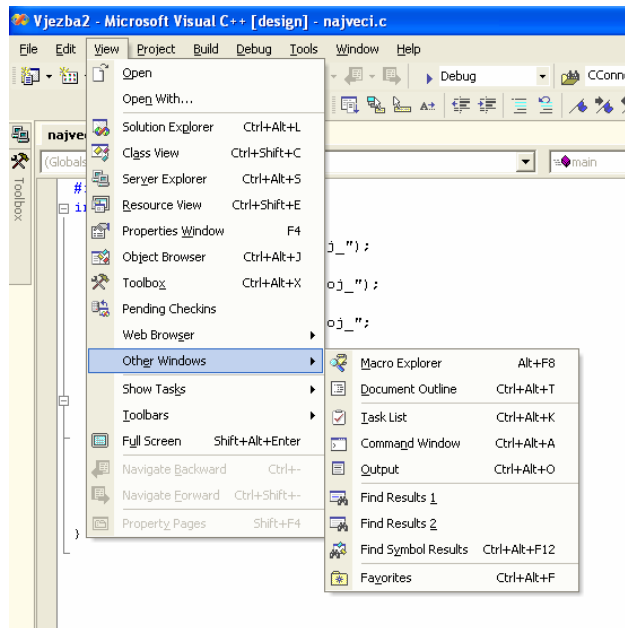
korištene operacije poput snimanja trenutno otvorene datoteke, ispisa na pisač i sl. Najčešće u dnu ekrana se nalaze prozori Task List i Output koji ispisuju poruke prevodioca i daju mogućnost pozicioniranja na pojedinu grešku unutar programa. U prozoru projekta možete vidjeti sve projekte koji su uključeni u vaš radni prostor (*Solution Explorer*) i sve datoteke koje pojedini projekt sadrži. Prozor projekta može sadržavati više kartica od kojih najčešće vidimo dvije kartice *Class View* i *Solution Explorer*. Kartica *Class View* sadrži podatke vezane uz programski jezik iz .NET skupa programskih jezika i nećete ju koristiti na ovim vježbama. Prozor za uređivanje teksta služi, kao što mu ime govori, za uređivanje teksta. Svaka otvorena datoteka unutar prozora za uređivanje teksta bit će prikazana kao zasebna kartica sa imenom datoteke na vrhu (u primjeru na slici to je samo jedna s imenom najveci.c). Osim označenih prozora sučelje može sadržavati i razne druge kao što je npr. prozor za poruke uz operacije koje razvojno okruženje obavlja (npr. pretraživanje datoteka).



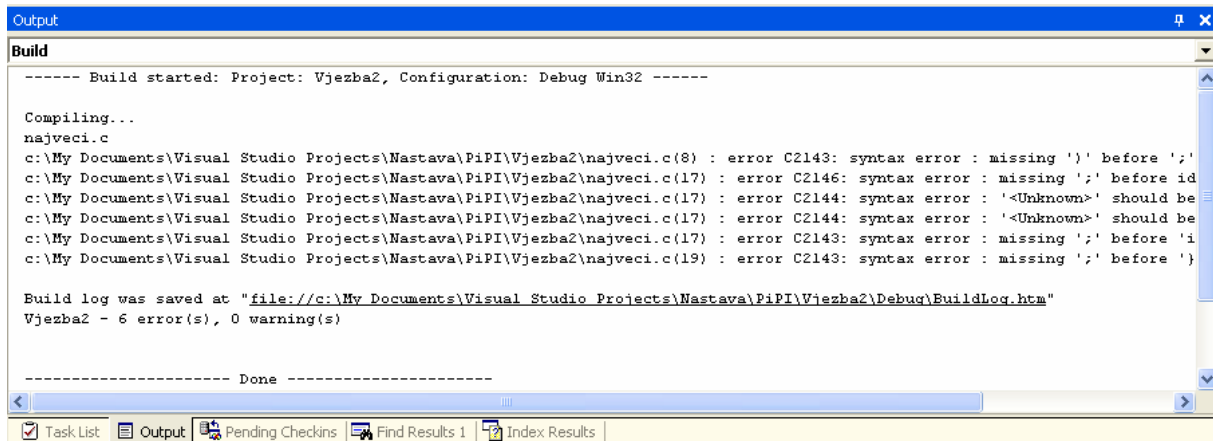
Slika 2.1 (tipičan izgled sučelja)

Sve prozore možete, po potrebi, zatvoriti pritiskom na znak X u gornjem desnom kutu pojedinog prozora. Ukoliko ponovno budete trebali prikaz nekog prozora možete ga uključiti kroz izbornik *View* kao što se vidi na slici 2.2.

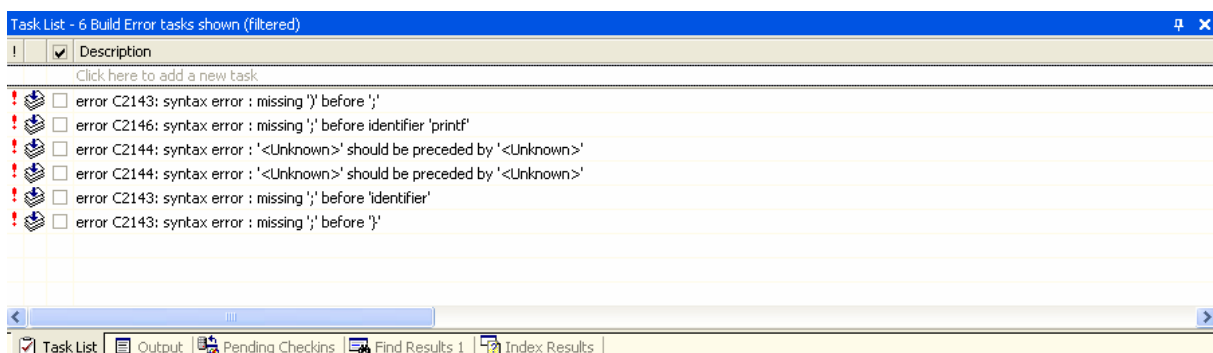
Ukoliko prema uputama pokušamo prevesti uneseni program (izbornik *Build* → *Build Vježba2*) uvidjet ćemo da probni program ima nekoliko sintaktičkih pogreški što će se prikladno ispisati u *Output* prozoru (slika 2.3) te će se svaka od grešaka pojaviti u *Task List* prozoru. (slika 2.4)



Slika 2.2 (izbornik View)



Slika 2.3 (prozor Output prikazuje poruke prevodioca prilikom prevođenja programa)



Slika 2.4 (prozor Task List sadrži listu pogrešaka prilikom prevođenja)

Dvostrukim klikom miša na pojedinu grešku unutar prozora Task List pozicionirat ćete se na redak u programu u kojem se nalazi ta pogreška.

Pokušajte sami ispraviti vaš program i ponovo ga prevesti. Ukoliko se u Output prozoru pojavi tekst nalik ovome:

```
Vježba2 - 0 error(s), 0 warning(s)
```

```
----- Done -----
```

```
Build: 1 succeeded, 0 failed, 0 skipped
```

to će značiti da je program uspješno preveden, tj. da ne sadrži nikakve sintaktičke pogreške te da ga možete pokrenuti. To možete učiniti korištenjem opcije `Start Without Debugging` iz izbornik `Debug` ili pritiskom na tipke `CTRL+F5`.

## Korištenje prekidnih točaka

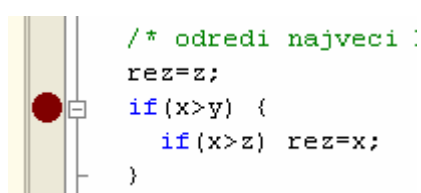
Osim pokretanja programa opcijom `Start Without Debugging`, program se može pokrenuti i u načinu rada za ispravljanje pogrešaka (`Debug mode`). Program se pokreće u načinu za ispravljanje pogrešaka odabirom opcije `Start` iz izbornika `Debug` ili pritiskom na tipku `F5`. (Napomena: ukoliko pokrenete program u ovom načinu rada, nakon završetka programa prozor sa pokrenutim programom će se automatski zatvoriti)

Ukoliko je program pokrenut u načinu za ispravljanje pogrešaka te ukoliko negdje unutar programa postoji prekidna točka (breakpoint) program će na tom mjestu privremeno zaustaviti svoje izvršavanje i čekati na našu daljnu akciju.

Postavljanje prekidne točke vrši se pozicioniranjem na liniju u kojoj želimo postaviti prekidnu točku i jednim od sljedeća tri načina:

- desni klik miša uzrokuje pojavljivanje 'pop-up' izbornika na kojem odaberemo opciju `Insert Breakpoint`
- klikom na zasivljeni dio sučelja lijevo od teksta
- odabirom opcije `New Breakpoint` iz izbornika `Debug`

Ukoliko je prekidna točka postavljena u nekoj liniji, na zasivljenom dijelu lijevo od teksta linije pojavit će se crveni krug.



Slika 2.5 (primjer postavljene prekidne točke)

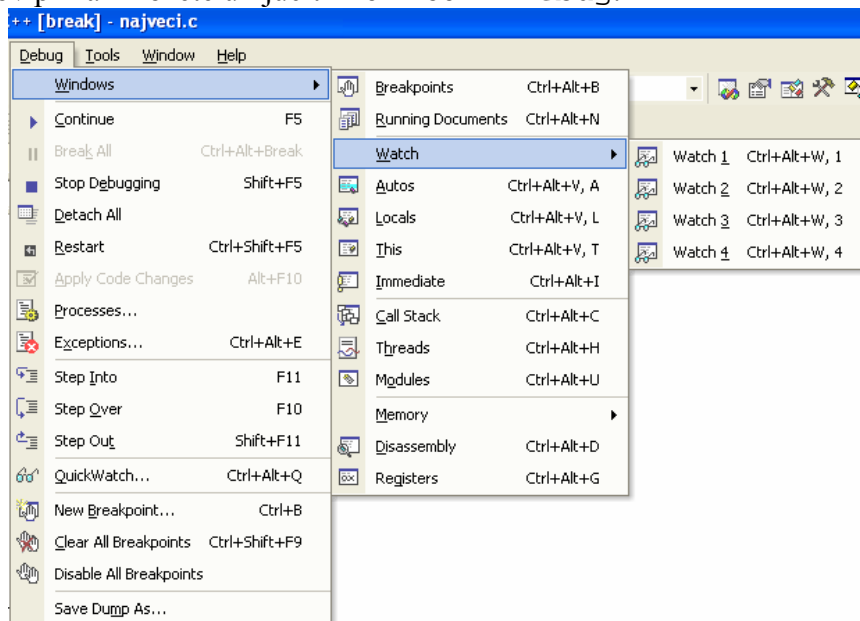
Pojedinu prekidnu točku možemo izbrisati tako da se pozicioniramo na liniju u kojoj se nalazi prekidna točka i odabemo jedan od sljedećih načina:

- desnim klikom miša pojavljuje se 'pop-up' izbornik s opcijom `Remove Breakpoint`
- klikom na zasivljeni dio sučelja lijevo od teksta nestaje crveni krug i prekidna točka je time izbrisana

Ukoliko je to potrebno, sve prekidne točke možemo izbrisati odabirom opcije `Clear All Breakpoints` iz izbornika `Debug`.

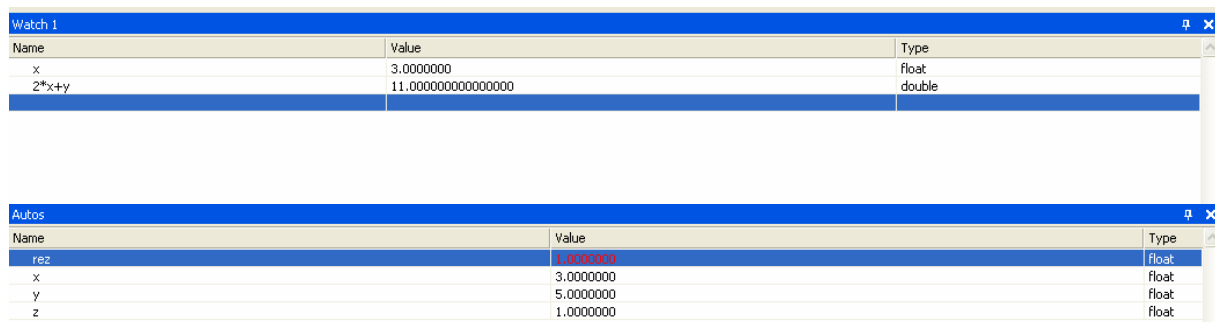
Isprobajte dodavanje prekidnih točaka, a zatim ostaviti jednu prekidnu točku i to u liniji koja počinje s `if (x>y)`.

Pokrenite program u načinu za ispravljanje pogrešaka. Nakon što je program došao do prekidne točke, njegovo je izvršavanje privremeno zaustavljeno. U tom trenutku možete pogledati vrijednosti varijabli koje vas zanimaju. Vrijednost varijabli možete vidjeti unutar prozora Autos i Watch1 (najčešće se nalaze na dnu ekrana). Ukoliko ovi prozori nisu vidljivi, njihov prikaz možete uključiti kroz izbornik Debug.



Slika 2.6 (uključivanje prikaza prozora Watch)

Prozor Autos automatski prikazuje vrijednosti varijabli koje se koriste u danom programskom odsječku, dok kod prozora Watch možete sami odrediti za koje varijable želite vidjeti vrijednost, a možete unijeti i neke jednostavne aritmetičke izraze koji koriste dane varijable.



Slika 2.7 (prozori Watch i Autos)

Vaš je zadatak da, u trenutku kad je program u prekidnoj točki, unutar Watch1 prozora dodate prikaz vrijednosti varijabli x,y,z i rez.

Nakon što ste promotrili vrijednosti varijabli u prekidnoj točki, nastavite izvršavanje programa redak po redak koristeći opciju Step Over iz izbornika Debug (isto možete izvesti pritiskom na tipku F10).

Osim ove opcije, u trenutku privremenog prekida programa zbog prekidne točke, korisne opcije izbornika Debug su:

- `Continue` : Nastavlja izvođenje programa od one točke u kojoj je izvršavanje programa zaustavljeno.
- `Stop Debugging` : Prekida izvođenje programa i izlazi iz načina rada za ispravljanje pogrešaka.
- `Restart` : Ponovno pokreće program i izlazi iz načina rada za ispravljanje pogrešaka
- `Step Into` : Izvodi sljedeću izvršnu naredbu programa. Ukoliko je sljedeća naredba poziv potprograma, izvođenje programa se zaustavlja na prvoj izvršnoj naredbi u tom potprogramu.
- `Step Over` : Izvodi sljedeću izvršnu naredbu programa. Ukoliko je sljedeća naredba poziv potprograma potprogram se izvede, a izvođenje se zaustavlja na sljedećoj izvršnoj naredbi iza tog poziva potprograma.
- `Step Out` : Izvodi sve naredbe do kraja potprograma koji se trenutno izvršava. Izvođenje se zaustavlja na prvoj izvršnoj naredbi iza poziva tog potprograma.