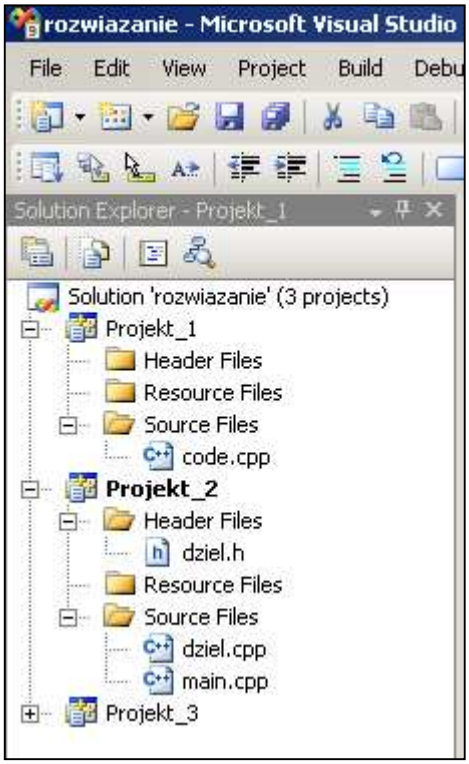
Laboratorium 9 – Zarządzanie projektami w Visual Studio

1. WIELE PROJEKTÓW W JEDNYM ROZWIĄZANIU



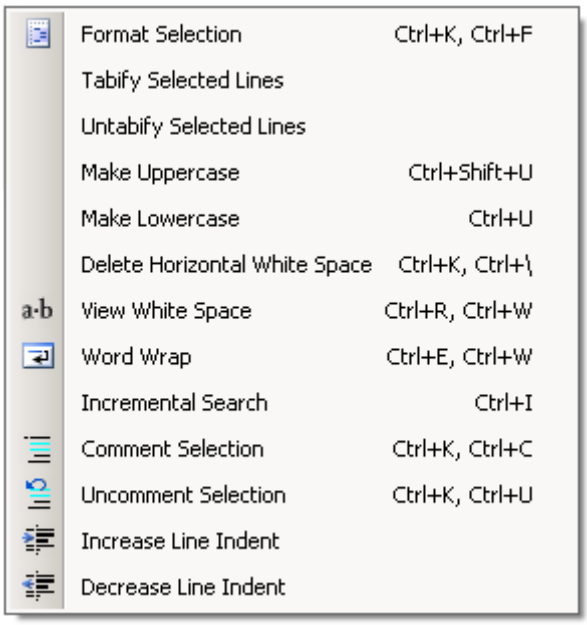
Nowy projekt:
File > New > Project > …
Nowy projekt konsolowy:
… > Visual C++ > Win32 > Win32 Console Application
Nowy projekt okienkowy:
… > Visual C++ > CLR > Windows Forms Application

Dodawanie nowego pliku źródłowego do projektu:
Kliknij PPM na Source Files > Add > New Item > Visual C++ > Code > C++ File (.cpp)
Dodawanie nowego pliku nagłówkowego do projektu:
Kliknij PPM na Header Files > Add > New Item > Visual C++ > Code > Header File (.h)

Okno 'Solution Explorer' - lista plików/projektów:
View > Solution Explorer
Nowy projekt w tym samym rozwiązaniu (Solution):
W oknie 'Solution Explorer' kliknij PPM na Solution > Add > New Project
Zmiana projektu domyślnego danego Solution:
Kliknij PPM na nazwę projektu > Set as StartUp Project
(aktualnie domyślny projekt zostanie wytłuszczony)

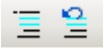
2. FORMATOWANIE KODU

Edit > Advanced > …



Automatyczne formatowanie kodu:
Zaznacz kod > Edit > Advanced > Format Selection

Zakomentuj / odkomentuj zaznaczony kod:



Zmniejsz / zwiększ wcięcie zaznaczonego kodu:

Zaznacz wszystko: Ctrl+A

PPM - prawy przycisk myszy,
LPM - lewy przycisk myszy

3. DEBUGGER

Breakpointy ustawia się na bocznej belce po lewej stronie edytora.

Breakpoint zwykły – zatrzymanie wykonania programu w danej linii.
Kliknij LPM na belce. Aby go usunąć, kliknij LPM jeszcze raz.

Breakpoint warunkowy – zatrzymanie wykonania programu w danej linii pod pewnym warunkiem.
Kliknij LPM na belce. Pojawi się zwykły breakpoint. Kliknij go teraz PPM > Condition > (warunek) Is true / Has changed, np. warunek *when 'a<b' is true.*

Breakpoint informacyjny – wyświetlenie komunikatu bez zatrzymania wykonania programu.
Kliknij LPM na belce. Pojawi się zwykły breakpoint. Kliknij go teraz PPM > When hit > Print a message (komunikat), np. komunikat *a={a}, b={b} Function: $FUNCTION, Thread: $TID $TNAME*

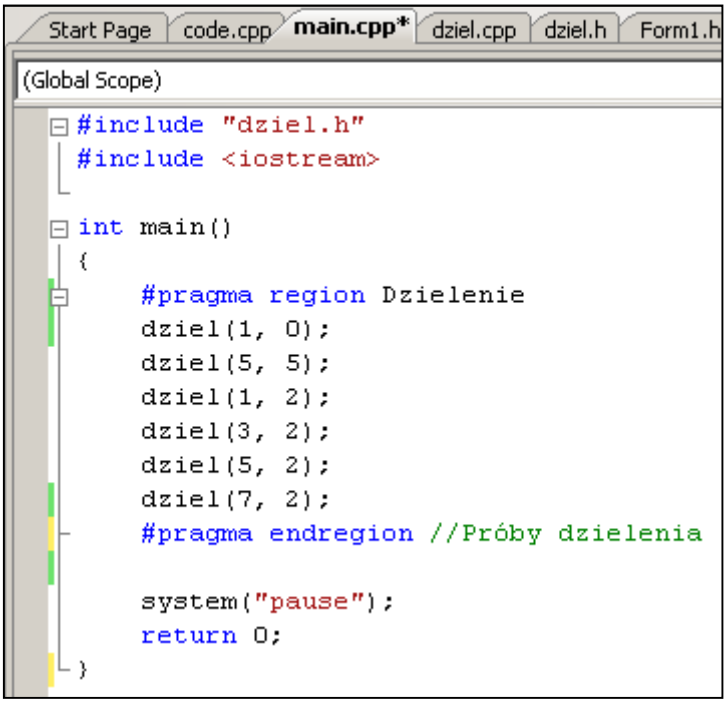
Podczas zatrzymanie w breakpoincie mamy podgląd wartości zmiennych (zakładka Locals) i wyjścia Output, gdzie pojawiają się komunikaty. Aby wyjść z breakpointa i tym samym pozwolić na dalsze wykonanie programu, kliknij F5 (Continue – zielona strzałka). Aby przerwać debugowanie w dowolnym momencie, kliknij Stop Debugging (niebieski kwadracik).

4. WYDZIELENIE REGIONU

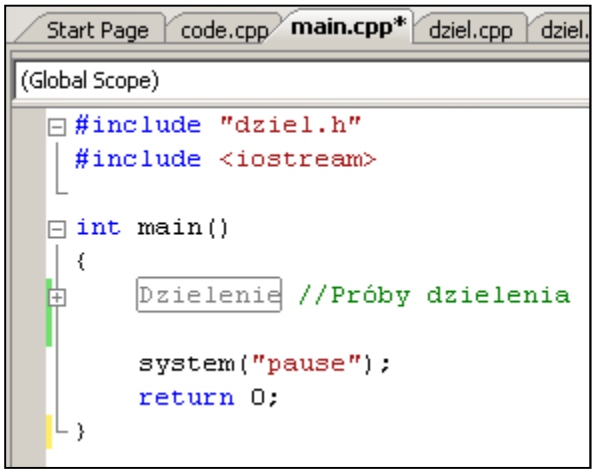
Składnia:

```
#pragma region Nazwa
      /* region o etykiecie Nazwa*/
#pragma endregion
```

W wersji rozwiniętej:



W wersji zwiniętej:



Zad. 1

Stwórz w jednym rozwiązaniu dwa projekty konsolowe ‘Projekt_1’ i ‘Projekt_2’ oraz jeden okienkowy ‘Projekt_3’.

Zad. 2

DEBUGGING

W projekcie ‘Projekt_1’ stwórz plik źródłowy code.cpp i ustaw następująco breakpointy:

Start Page | code.cpp | main.cpp | dziel.cpp | dziel.h | Form1.h [Design]

(Global Scope) | main()

```
#include <iostream>
using namespace std;

double dziel(int a, int b)
{
    if(b==0){
        cout << "Wynik dzielenia " << a << "/" << b << " = " << RAND_MAX << endl;
        return RAND_MAX;
    }
    else{
        cout << "Wynik dzielenia " << a << "/" << b << " = " << a*1.0/b << endl;
        return a*1.0/b;
    }
}

int main()
{
    dziel(1, 0);
    dziel(3, 2);
    dziel(5, 2);
    dziel(7, 10);
    dziel(9, 3);

    cout<<"\n przed \n";
    cout<<"\n po \n\n";

    system("pause");
    return 0;
}
```

Wpisz odpowiedni komunikat i warunek. Zdebuguj projekt.

Zad. 3

BIBLIOTEKA STATYCZNA

W projekcie 'Projekt_2' stwórz pliki dziel.cpp i main.cpp oraz plik nagłówkowy dziel.h:

```
//dziel.cpp
#include "dziel.h"
#include <iostream>
using namespace std;

double dziel(int a, int b)
{
   if(b==0){
      cout << "Wynik dzielenia " << a << "/" << b << " = " << RAND_MAX << endl;
      return RAND_MAX;
   }
   else{
      cout << "Wynik dzielenia " << a << "/" << b << " = " << a*1.0/b << endl;
      return a*1.0/b;
   }
}
```

```
//dziel.h
#ifndef DZIEL
#define DZIEL

double dziel(int, int);

#endif
```

```
//main.cpp
#include "dziel.h"
#include <iostream>

int main()
{
   dziel(1, 0);
   dziel(3, 2);
   dziel(5, 2);
   dziel(7, 10);
   dziel(9, 3);

   system("pause");
   return 0;
}
```

Ustaw 'Projekt_2' jako projekt domyślny i skompiluj go.

W pliku main.cpp

- kliknij PPM na funkcję dziel(), a następnie Go to Definition,
- kliknij PPM na funkcję dziel(), a następnie Go to Declaration.

Które pliki zostały otwarte?