



Łukasz Orzeszko

Dev C++



Czym jest Dev-C++?

DevC++ jest w pełni funkcjonalnym darmowym środowiskiem programistycznym C/C++ zawierającym wielookienkowy edytor kodu źródłowego z podświetlaniem składni, kompilator, debbuger, linker.



Co to jest kompilator?

Kompilator – program służący do automatycznego tłumaczenia kodu napisanego w jednym języku (języku *źródłowym*) na równoważny kod w innym języku (języku *wynikowym*)

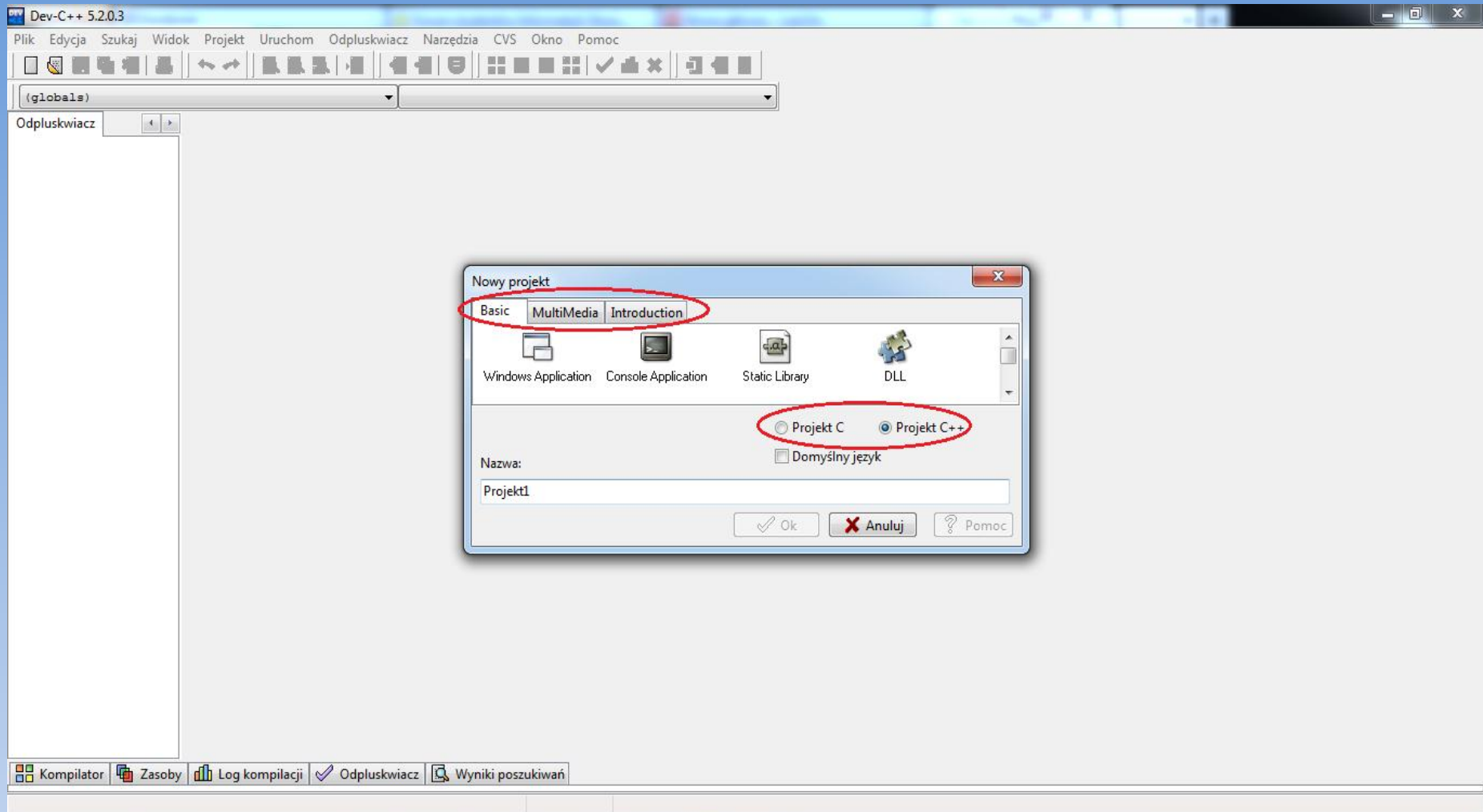


Co to jest debugger?

Debugger - program komputerowy służący do dynamicznej analizy innych programów, w celu odnalezienia i identyfikacji zawartych w nich błędów, zwanych z angielskiego **bugami** (robakami). Proces nadzorowania wykonania programu za pomocą debuggera określa się mianem debugowania.

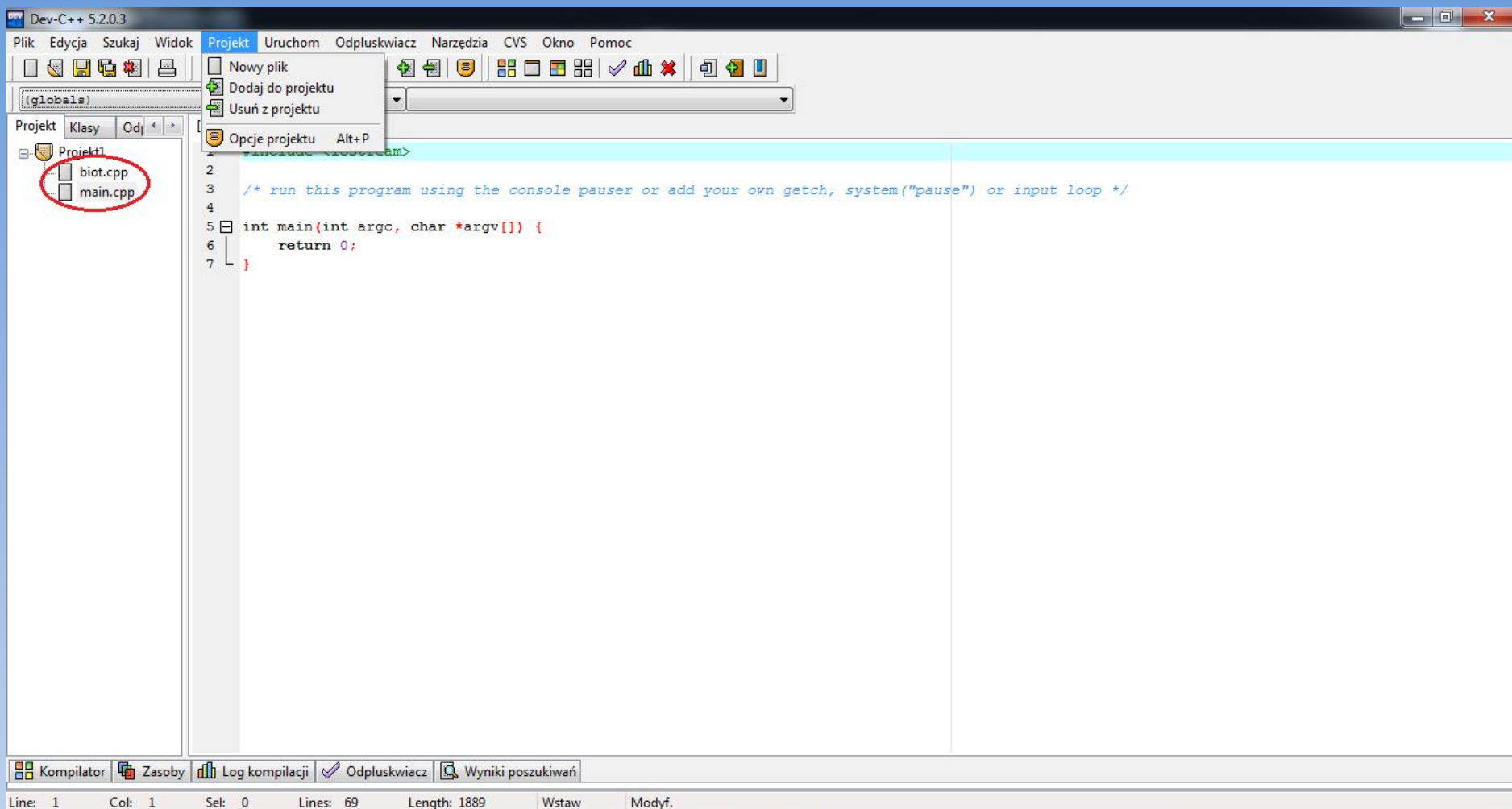


Tworzenie projektu



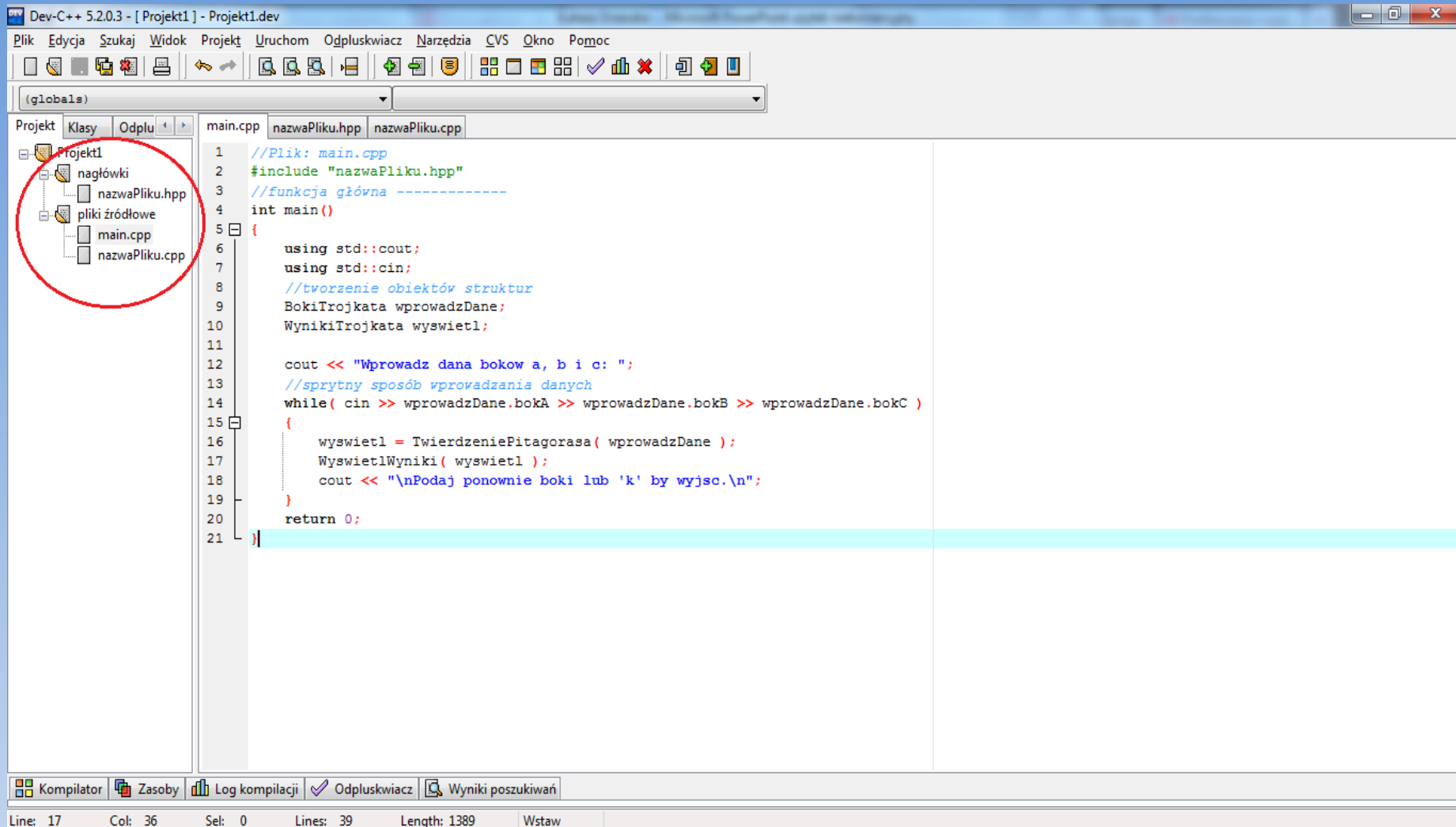


Tworzenie projektu





Tworzenie projektu





Edycja i kompilacja

Dev-C++ 5.2.0.3 - [Projekt1] - Projekt1.dev

Plik Edycja Szukaj Widok Projekt Uruchom Odpluskwiacz Narzędzia CVS Okno Pomoc

(globals)

Projekt Klasy Odplu

Projekt1

- nagłówki
- nazwaPliku.hpp
- pliki źródłowe
- main.cpp
- nazwaPliku.cpp

main.cpp

```
1 //Plik: main.cpp
2 #include "nazwaPliku.hpp"
3 //funkcja główna -----
4 int main()
5 {
6     using std::cout;
7     using std::cin;
8     //tworzenie obiektów struktur
9     BokiTrojkata wprowadzDane;
10    WynikiTrojkata wyswietl;
11
12    cout << "Wprowadz dane bokow a, b i c: ";
13    //sprytny sposób wprowadzania danych
14    while( cin >> wprowadzDane.bokA >> wprowadzDane.bokB >> wprowadzDane.bokC )
15    {
16        wyswietl = TwierdzeniePitagorasa( wprowadzDane );
17        WyswietlWyniki( wyswietl );
18        cout << "\nPodaj ponownie boki lub 'k' by wyjsc.\n";
19    }
20    return 0;
21 }
```

Opcje edytora

Ogólne Widok Kolorowanie składni Kod Przeglądarka klas Autosave

Assembler
Character
Comment
Float
Hexadecimal
Identifier
Illegal Char
Number
Octal

Kolor tekstu:

Kolor tła:

Styl:

- ☐ Pogrubiony
- ☐ Kursywa
- ☐ Podkreślony

Szybki wybór stylu:

☒ Używaj kolorowania składni

c:\cpp\h\hpp;cc;co;cp;hp;rh;fx;inl;

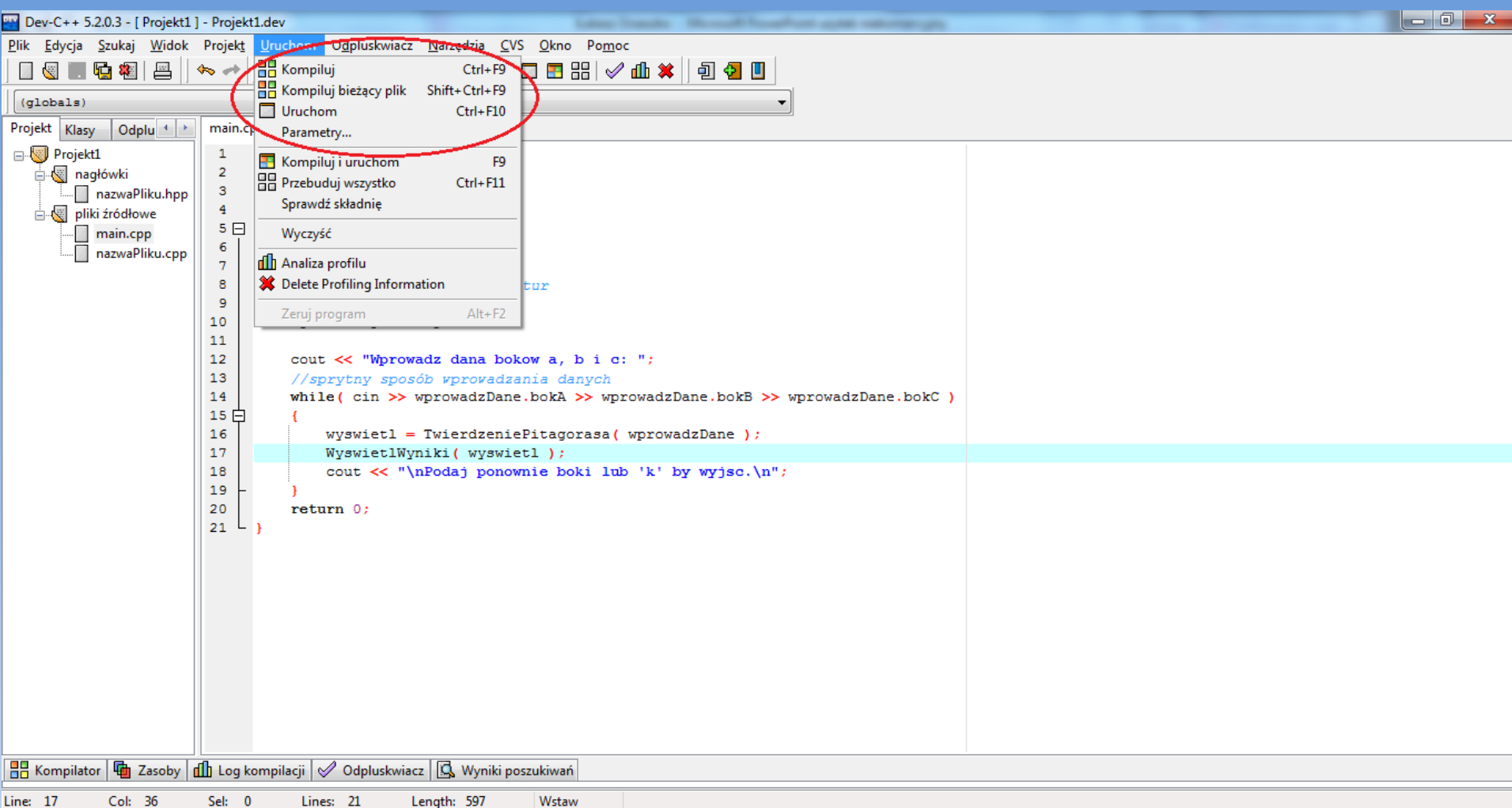
Ok Anuluj Pomoc

Kompilator Zasoby Log kompilacji Odpluskwiacz Wyniki poszukiwań

Line: 17 Col: 36 Sel: 0 Lines: 21 Length: 597 Wstaw

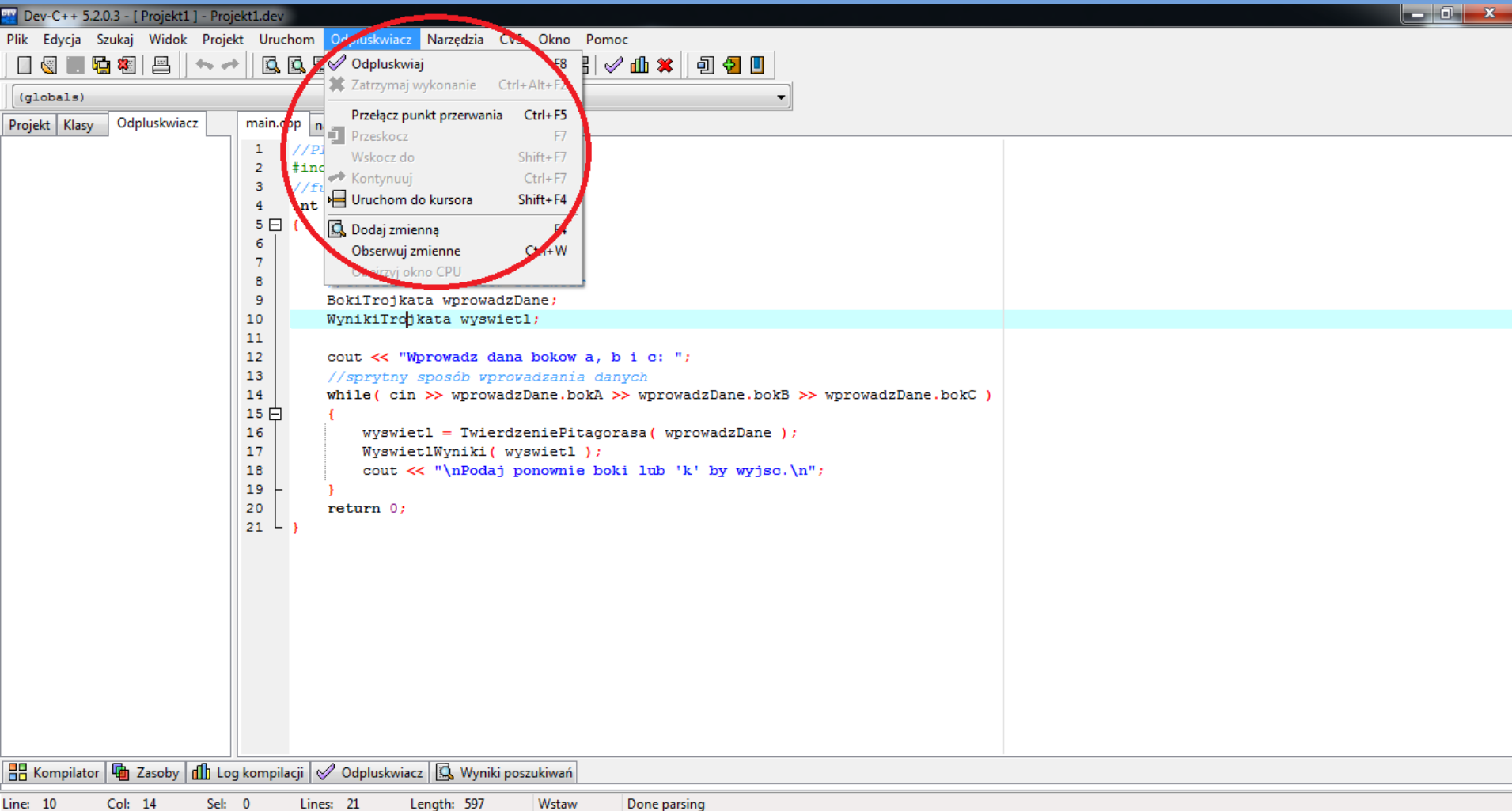


Edycja i kompilacja





Debuggowanie





Debuggowanie

Dev-C++ 5.2.0.3 - [Projekt1] - Projekt1.dev

Plik Edycja Szukaj Widok Projekt Uruchom Odpluskwiacz Narzędzia CVS Okno Pomoc

(globals)

Projekt Klasy Odpluskwiacz

wprowadzDane
bokA = 1.796145478025017e-312
bokB = 5.0553161699131657e+264
bokC = 2.0872530087822935e-314
czy_prawda = false
TwierdzeniePitagorasa = {WynikiTrojkata
cout = <incomplete type>
main = ?
BokiTrojkata = ?
WynikiTrojkata = ?
WyswietlWyniki = {void (const Wyniki
wyswietl
Wynik1 = 6.9520656385972404e-308
Wynik2 = 6.9531011709949285e-308
Wynik3 = -5.1669857364743543e-24
czy_prawda = 254
BokiTrojkata = ?
std = ?
std = Could not watch this variable
\$eax = 0x1
\$ebx = 0x7efde000
\$ecx = 0x1
\$edx = 0x2
\$esi = 0x0
\$edi = 0x0

main.cpp nazwaPliku.hpp nazwaPliku.cpp

```
1 //Plik: main.cpp
2 #include "nazwaPliku.hpp"
3 //funkcja główna -----
4 int main()
5 {
6     using std::cout;
7     using std::cin;
8     //tworzenie obiektów struktur
9     BokiTrojkata wprowadzDane;
10    WynikiTrojkata wyswietl;
11
12    cout << "Wprowadz dane bokow a, b i c: ";
13    //sprytny sposób wprowadzania danych
14    while( cin >> wprowadzDane.bokA >> wprowadzDane.bokB >> wprowadzDane.bokC )
15    {
16        wyswietl = TwierdzeniePitagorasa( wprowadzDane );
17        WyswietlWyniki( wyswietl );
18        cout << "\nPodaj ponownie boki lub 'k' by wyjsc.\n";
19    }
20    return 0;
21 }
```

Kompilator Zasoby Log kompilacji Odpluskwiacz Wyniki poszukiwań Zamknij

Odpluskwiacz Śledzenie wsteczne Wyjście

Przeskocz Wskocz do Kontynuuj Uruchom do kursora Odpluskwiacz Zatrzymaj wykonanie Dodaj zmienną Usuń zmienną

Line: 12 Col: 1 Sel: 0 Lines: 21 Length: 597 Wstaw Done parsing



Debuggowanie

Dev-C++ 5.2.0.3 - [Projekt1] - Projekt1.dev

Plik Edycja Szukaj Widok Projekt Uruchom

Odpluskwiacz Narzędzia CVS OK

(globals)

Projekt Klasy Odpluskwiacz

- wprowadzDane
- bokA = ?
- bokB = ?
- bokC = ?
- czy_prawda = ?
- TwierdzeniePitagorasa = ?
- cout = ?
- main = ?
- BokiTrojkata = ?
- WynikiTrojkata = ?
- WyswietlWyniki = ?
- wyswietl
- Wynik1 = ?
- Wynik2 = ?
- Wynik3 = ?
- czy_prawda = ?
- BokiTrojkata = ?
- std = ?
- std = Could not watch this variable
- Seax = 0x1
- Sebx = 0x7efde000
- Secx = 0x1
- Sedx = 0x2
- Sesi = 0x0
- Sedi = 0x0

Przełącz punkt przerwania Ctrl+F5

Przeskocz F7

Wskocz do Shift+F7

Kontynuuj Ctrl+F7

Uruchom do kursora Shift+F4

Dodaj zmienną F4

Obejrzyj okno CPU

9 BokiTrojkata wprowadzDane

10 WynikiTrojkata wys

11

12 cout << "Wprowadz

13 //sprytny sposób w

14 while(cin >> wpro

15 {

16 wyswietl = Twi

17 WyswietlWyniki

18 cout << "\nPod

19 }

20 return 0;

21 }

Okno CPU

Kod asemblera:

Funkcja: main()

```
0x004013cc <+0>: lea 0x4(%esp),%ecx
0x004013d0 <+4>: and $0xffffffff0,%esp
0x004013d3 <+7>: pushl -0x4(%ecx)
0x004013d6 <+10>: push %ebp
0x004013d7 <+11>: mov %esp,%ebp
0x004013d9 <+13>: push %edi
0x004013da <+14>: push %esi
0x004013db <+15>: push %ebx
0x004013dc <+16>: push %ecx
0x004013dd <+17>: sub $0xa8,%esp
0x004013e3 <+23>: call 0x41c124 <mcount>
0x004013e8 <+28>: call 0x4012bc <_monstartup>
0x004013ed <+33>: call 0x40e838 <_main>
=> 0x004013f2 <+38>: movl $0x478064,0x4(%esp)
0x004013fa <+46>: movl $0x476fc0,(%esp)
0x00401401 <+53>: call 0x4698cc <_ZStlsISt11char_traitsIcEERSt13basic_os
0x00401406 <+58>: jmp 0x40146e <main()+162>
0x00401408 <+60>: lea -0x78(%ebp),%eax
0x0040140b <+63>: mov %eax,-0x7c(%ebp)
0x0040140e <+66>: lea 0x4(%esp),%edx
0x00401412 <+70>: lea -0x58(%ebp),%ebx
0x00401415 <+73>: mov $0x8,%eax
0x0040141a <+78>: mov %edx,%edi
0x0040141c <+80>: mov %ebx,%esi
0x0040141e <+82>: mov %eax,%ecx
0x00401420 <+84>: rep movsl %ds:(%esi),%es:(%edi)
0x00401422 <+86>: mov -0x7c(%ebp),%eax
0x00401425 <+89>: mov %eax,%esi
```

Składnia asemblera:

AT&T Intel

Rejestry:

EAX: 0x1

EBX: 0x7efde000

ECX: 0x1

EDX: 0x2

ESI: 0x0

EDI: 0x0

EBP: 0x28ff18

ESP: 0x28fe60

EIP: 0x4013f2

CS: 0x23

DS: 0x2b

SS: 0x2b

ES: 0x2b

FS: 0x53

GS: 0x2b

Kompilator Zasoby Log kompilacji Odpluskwiacz Wyniki poszukiwań Zamknij

Odpluskwiacz Śledzenie wsteczne Wyjście

Przeskocz Kontynuuj

Wskocz do Uruchom do kursora

Odpluskwiacz

Zatrzymaj wykonanie

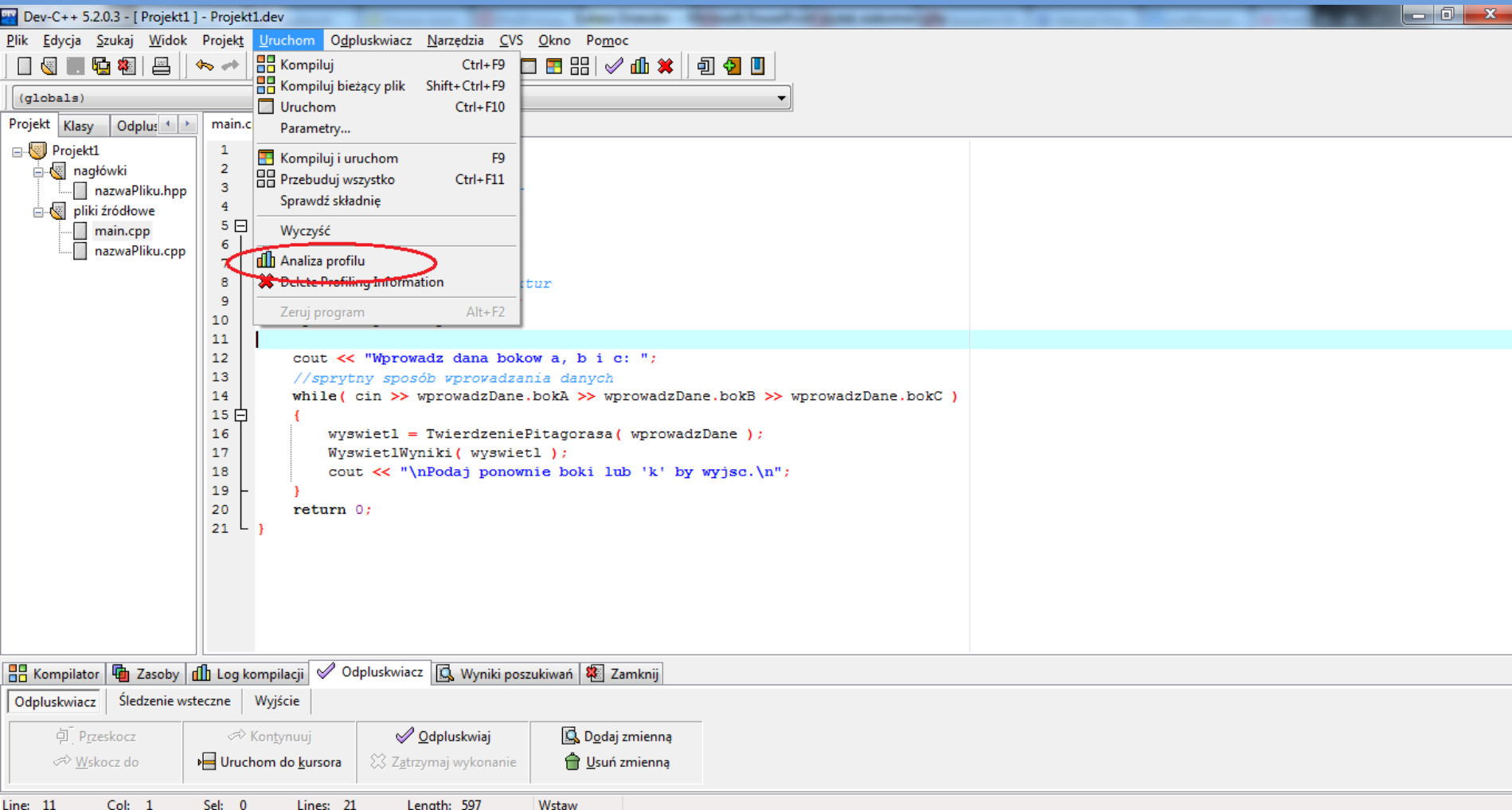
Dodaj zmienną

Usuń zmienną

Line: 12 Col: 1 Sel: 0 Lines: 21 Length: 597 Wstaw Done parsing



Profilowanie





Profilowanie

Dev-C++ 5.2.0.3 - [Projekt1] - Projekt1.dev

Plik Edycja Szukaj Widok Projekt Uruchom Odpluskwiacz Narzędzia CVS Okno Pomoc

(globals)

Projekt Klasy Odplu:

Projekt1
nagłówki
nazwaPliku.hpp
pliki źródłowe
main.cpp
nazwaPliku.cpp

main.cpp

```
1 //Plik: main.cpp
2 #include "nazwaPliku.hpp"
3 //funkcja główna
4 int main()
5 {
6     using std::cout;
7     using std::endl;
8     //tworzenie
9     BokiTrojkata
10    WynikiTrojkata
11
12    cout << "Wprowadź dane dla trójkąta:\n";
13    //sprytny sposób
14    while( cin > 0 )
15    {
16        wyswietl
17        Wyswietl
18        cout << endl;
19    }
20    return 0;
21 }
```

Analiza profilu

Plaskie wyjście Graf wywołań Profiling Options

Function name	% time	Cumul. se...	Self secs	Calls	Self ts/call	Total ts/call
std::pow(double, int)	0.00	0.00	0.00	6	0.00	0.00
WyswietlWyniki(WynikiTrojkata)	0.00	0.00	0.00	1	0.00	0.00
TwierdzeniePitagorasa(BokiTrojkata)	0.00	0.00	0.00	1	0.00	0.00

% the percentage of the total running time of the program used by this function.

cumulative a running sum of the number of seconds accounted for by this function and those listed above it.

self the number of seconds accounted for by this

Kompilator Zasoby Log kompilacji Odpluskwiacz Śledzenie wsteczne Wyjście

Przeskocz Kontynuuj Uruchom do kursora Wskocz do

Odpluskwiacz Dodaj zmienną Usuń zmienną

Line: 11 Col: 1 Sel: 0 Lines: 21 Length: 597 Wstaw



Czym jest MinGW?

MinGW (Minimalist GNU for Windows) – port GCC, dostarczający darmowe i otwarte środowisko oraz narzędzia pozwalające na kompilację natywnych plików wykonywalnych dla platformy Windows.



Paczki w ramach MinGW

binutils - Assembler, linker, ...

mingw-runtime - Headers and libraries for the C

library
w32api - Windows API header files and
libraries

gcc-core - C compiler

gcc-g++ - C++ compiler

gcc-objc - Objective C compiler

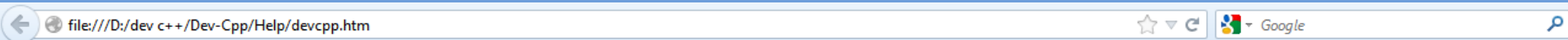
gcc-gfortran - Fortran 90/95 compiler

gcc-java - Java compiler

gcc-ada - Ada compiler



W razie czego...



Welcome to Orwell Dev-C++'s severely neglected help file! You can start navigating it from here.

User interface documentation

- Dialog Windows
 - Profile Analysis
 - [Flat Output](#)
 - [Call Graph](#)
 - [Profiling Options](#)
 - Project Options
 - [General](#)
 - [Files](#)
 - [Compiler](#)
 - [Parameters](#)
 - [Directories](#)
 - [Build Options](#)
 - [Makefile](#)
 - [Version Info](#)
- Start Menu items
 - [File](#)
 - [Edit](#)
 - [Search](#)
 - [View](#)
 - [Project](#)

Frequently Asked Questions



Koniec

Dziękuję za wysłuchanie prezentacji ;)