

# SỬ DỤNG CẤU TRÚC LẬP

## Một số thao tác Debug

### 1. Tạo project

- Đặt tên project **SimpleDebug\_proj**
- Đặt tên solution **SimpleDebug\_solu**
- Tạo file mã nguồn **SimpleDebug.cpp**

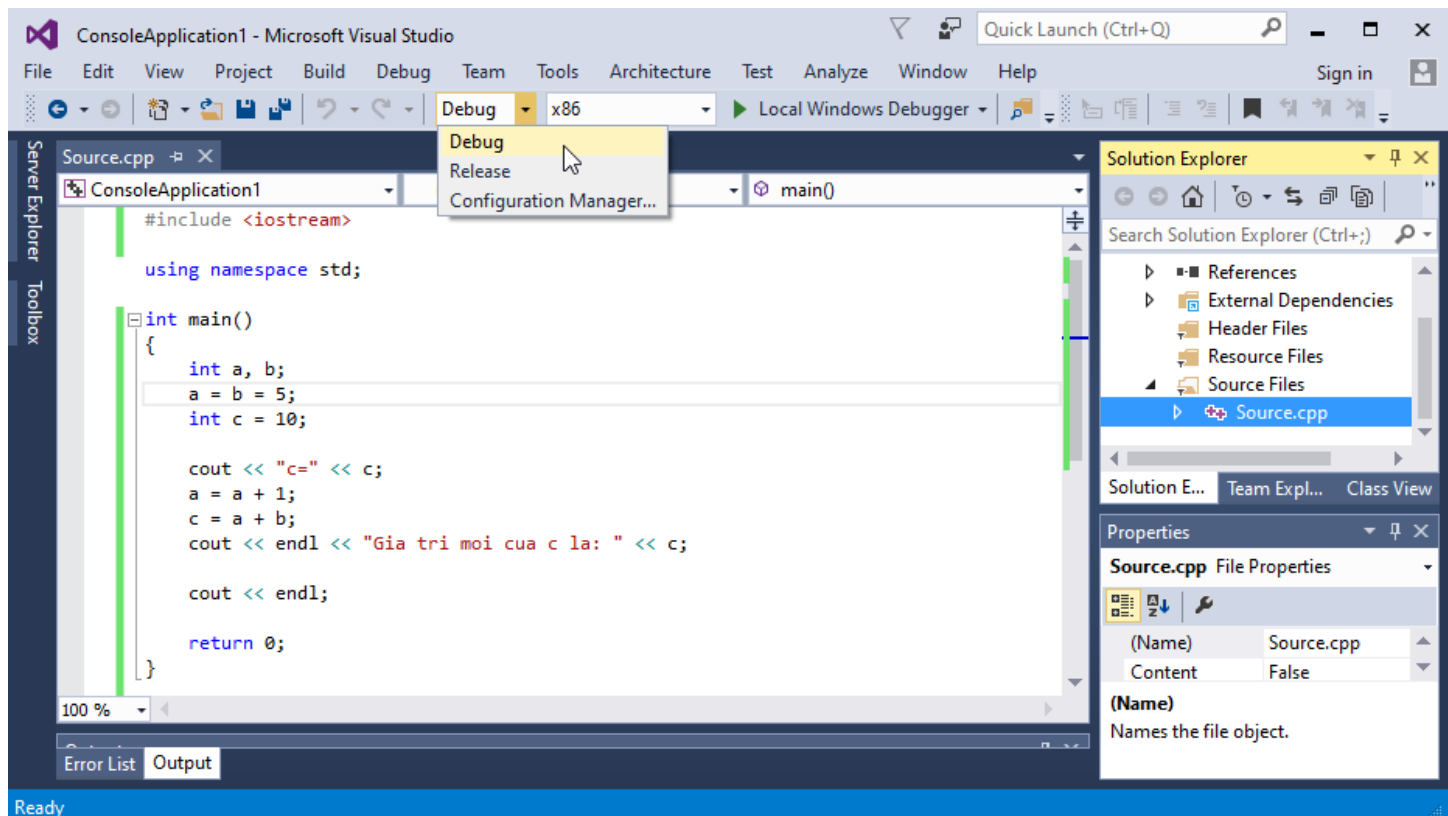
```
#include <iostream>
using namespace std;

int main()
{
    int a, b;
    a = b = 5;
    int c = 10;

    cout << "c=" << c;
    a = a + 1;
    c = a + b;
    cout << endl << "Gia tri moi cua c la: " << c;

    cout << endl;
    return 0;
}
```

### 2. Hai chế độ Build



### 3. Quan sát thư mục chương trình

- Thư mục solution

Name	Date modified	Type	Size
.vs	9/13/2018 8:40 PM	File folder	
Debug	9/13/2018 8:51 PM	File folder	
Release	9/13/2018 8:52 PM	File folder	
SimpleDebug_proj	9/13/2018 9:13 PM	File folder	
SimpleDebug_solu.sln	9/13/2018 8:40 PM	Microsoft Visual Studio Solution	2 KB
SimpleDebug_solu.VC.db	9/13/2018 8:40 PM	Data Base File	220 KB
SimpleDebug_solu.VC.VC.opendb	9/13/2018 8:40 PM	OPENDB File	0 KB

- Thư mục project

Name	Date modified	Type	Size
Debug	9/13/2018 9:13 PM	File folder	
Release	9/13/2018 8:52 PM	File folder	
SimpleDebug.cpp	9/13/2018 9:13 PM	C++ Source	1 KB
SimpleDebug_proj.vcxproj	9/13/2018 8:48 PM	VC++ Project	8 KB
SimpleDebug_proj.vcxproj.filters	9/13/2018 8:48 PM	VC++ Project Filte...	1 KB

### 4. Chạy chương trình từng lệnh

- Chọn chế độ Debug cho Solution Configurations
- Debug > Step Over (F10)
- Xem giá trị các biến qua cửa sổ **Autos** hoặc rê chuột trên từng biến.

ConsoleApplication1 (Debugging) - Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Architecture Test Analyze Window Help

Process: [5252] ConsoleApplication1.exe Lifecycle Events Thread: [8988] Main Thread

Source.cpp

```

int main()
{
    int a, b;
    a = b = 5;
    int c = 10; // ≤ 1ms elapsed

    cout << "c=" << c;
    a = a + 1;
    c = a + b;
    cout << endl << "Gia tri moi cua c la: " << c;
}

```

Autos

Name	Value	Type
a	5	int
b	5	int
c	-858993460	int

Call Stack

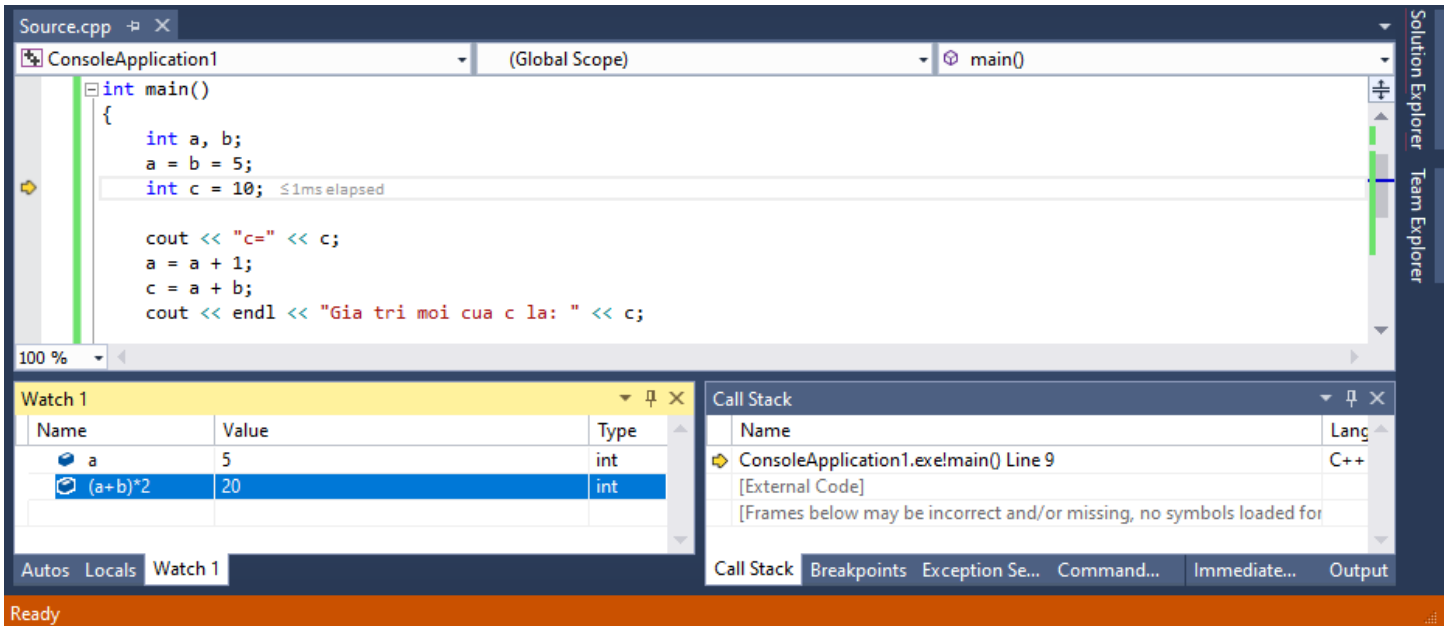
Name	Lang
ConsoleApplication1.exe!main() Line 9	C++
[External Code]	
[Frames below may be incorrect and/or missing, no symbols loaded for ...]	

Autos Locals Watch 1

Loading symbols for bcryptprimitives.dll

Ln 9 Col 1 Ch 1 INS

- Xem giá trị các biến/biểu thức qua cửa sổ **Watch**:
  - Debug > Window > Watch...
  - Nhập các biến/biểu thức trên từng dòng trong cửa sổ Watch.

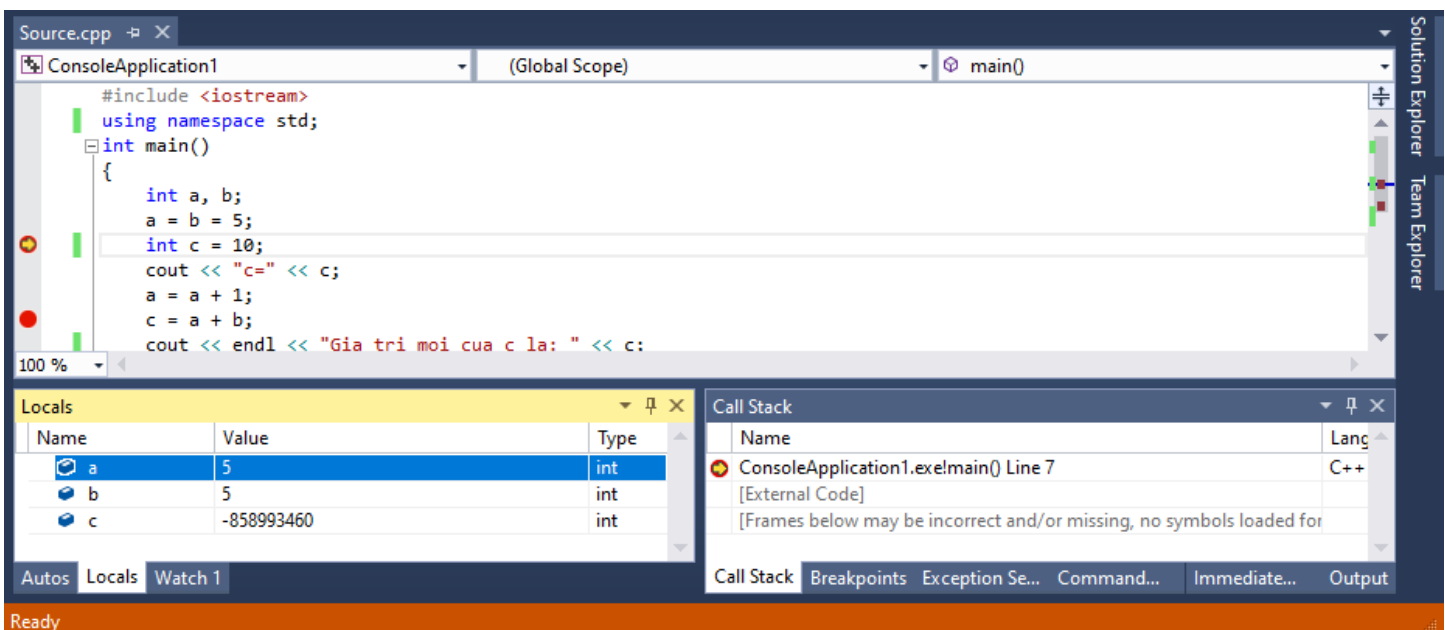


## 5. Kết thúc debugging:

- Debug > Stop Debugging (Shift+F5)

## 6. Đặt Breakpoints

- Chọn từng dòng lệnh muốn đặt breakpoint
- Debug > Toggle Breakpoint (F9)
- Chạy chương trình ở chế độ debugging: Debug > Start Debugging (F5)
- Nhấn F5 để đến breakpoint kế tiếp, hoặc nhấn F10 chạy từng lệnh.



7. **Tạo project Circle** với các lệnh trong hàm main() như dưới đây, sử dụng debugging để tìm và sửa lỗi chương trình.

```
float p, s;  
int r;  
cout << "Nhap vao ban kinh hinh tron: ";  
cin >> r;  
p = 2*PI*r;  
s = PI*r*r;  
cout << "Chu vi hinh tron: " << p << endl;  
cout << "Dien tich hinh tron: " << s << endl;
```

## CODE SAMPLE

Tính tổng các số chia hết cho 4 mà không chia hết cho 5 và nhỏ hơn **n**.

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    long s = 0;
    cout << "Nhap vao n: ";
    cin >> n;
    for (int i = 1; i <= n; i++)
        if (i % 4 != 0 && i % 5 == 0)
            s += i;
    cout << "Ket qua la: s= " << s;
    return 0;
}
```

Nhập vào 1 số nguyên dương, xuất ra số ngược lại, ví dụ nhập 123, xuất ra 321.

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    do{
        cout << "Nhap so nguyen duong n: ";
        cin >> n;
    } while (n <= 0);
    int don_vi = n % 10;
    while (don_vi != 0)
    {
        cout << don_vi;
        n = n / 10;
        don_vi = n % 10;
    }
    return 0;
}
```