

# HƯỚNG DẪN THỰC HÀNH TUẦN 3

## Một số thao tác Debug

### 1. Tạo project

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

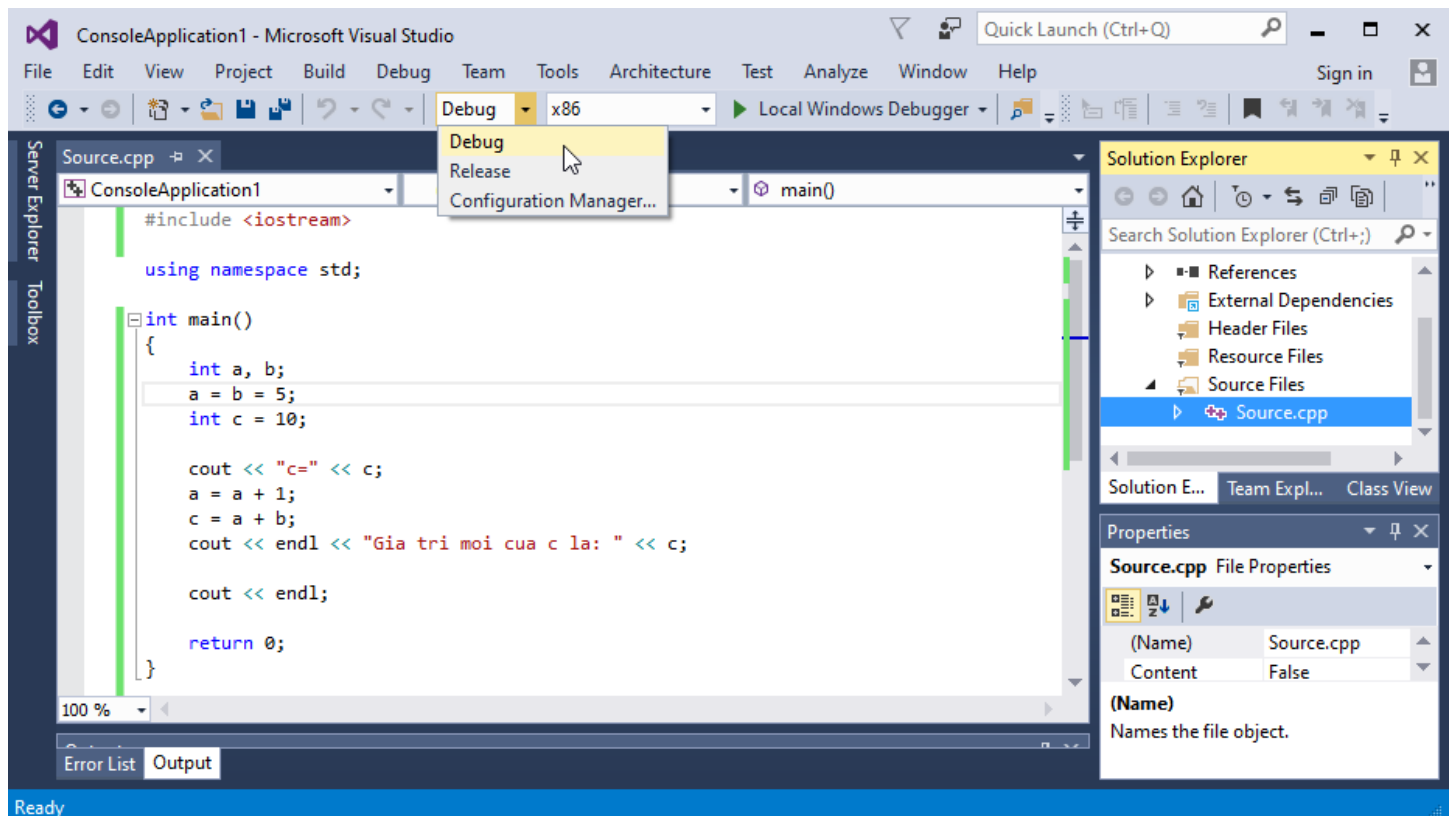
```
#include <iostream>
using namespace std;

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;

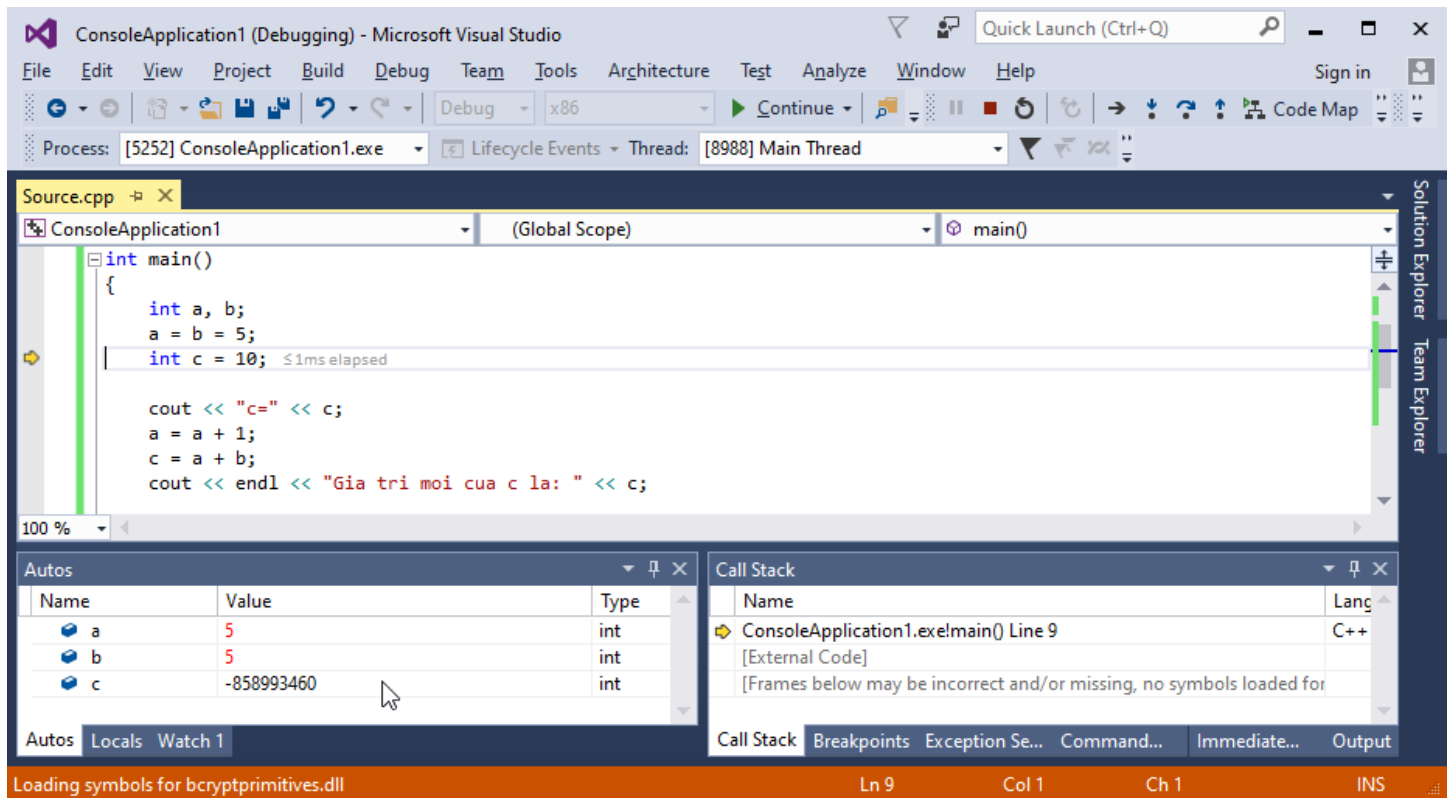
    cout << endl;
    return 0;
}
```

### 2. Hai chế độ Build

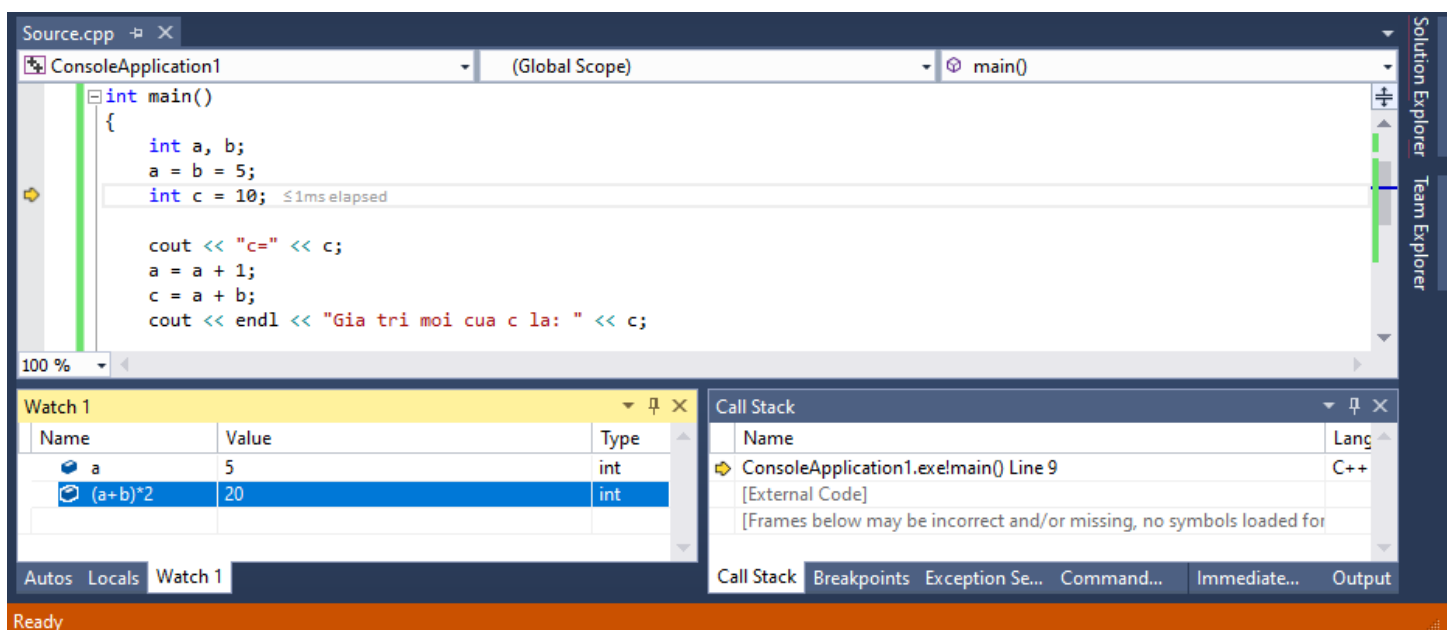


### 3. Chạy chương trình từng lệnh

- 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.



- 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.

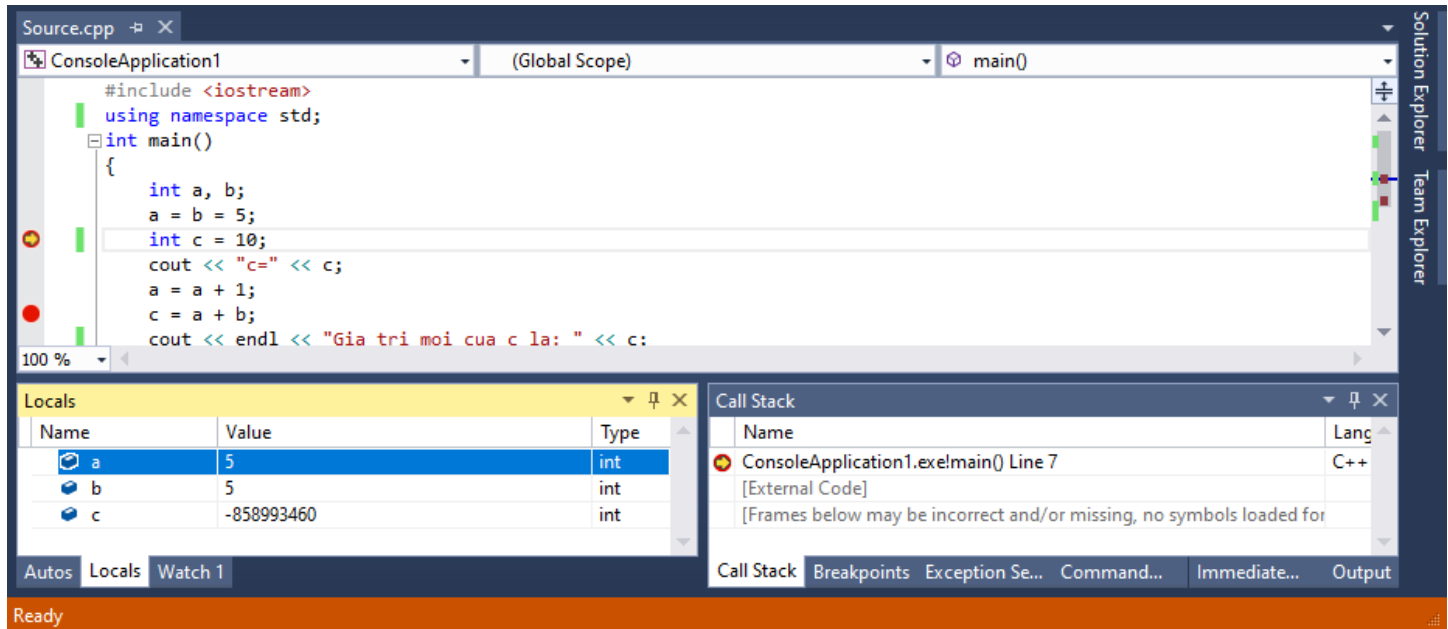


#### 4. Kết thúc debugging:

- Debug > Stop Debugging (Shift+F5)

#### 5. 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.



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

```
#include <iostream>
using namespace std;
#define PI 3.141592653589793
int main()
{
    double 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 tích hinh tron: " << s << endl;

    return 0;
}
```

## Một số ví dụ sử dụng cấu trúc lặp for, while và do..while

1. Tính tổng các số nguyên chia hết cho 4 mà không chia hết cho 5 và nhỏ hơn số nguyên **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;
}
```

2. 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;
}
```

## Bài tập thực hành

1. Nhập vào 1 số nguyên, tính tổng các chữ số của số nguyên đó.

Input: 543

Output: Tong cac chu so la 12

2. Nhập vào 1 số nguyên, in ra màn hình cách đọc số nguyên đó.

Input: 543

Output: Nam Bon Ba

3. Nhập vào 1 số nguyên dương, kiểm tra các chữ số của số nguyên dương có giảm dần hay tăng dần từ trái sang phải không.

Input: 5436

Output: Cac chu so khong tang/giam dan tu trai sang phai

4. Nhập vào 1 số nguyên dương, tìm chữ số lớn nhất và chữ số nhỏ nhất của số nguyên dương.

Input: 5436

Output: Chu so lon nhat la 6

Chu so nho nhat la 3

5. Nhập vào 1 số nguyên dương, liệt kê tất cả các ước số của số nguyên dương. Cho biết có bao nhiêu ước số và tìm tổng của tất cả các ước số đó.

Input: 16

Output: Cac uoc so la 1 2 4 8 16

Co 5 uoc so

Tong cac uoc so la 31

6. Nhập vào 1 số nguyên, kiểm tra số đó có phải là số nguyên tố không.

Input: 12

Output: Khong phai la so nguyen to

7. Nhập vào 2 số nguyên M và N, với  $M < N$ . Đếm số lượng các số chẵn trong  $[M, N]$  và tính tổng các số đó.

Input: 5 12

Output: So luong so chan la 4

Tong cac so chan la 36