Bài 5b:   
  
\_Ở bài tập này, chúng ta thực nghiệm với 2 kiểu dữ liệu là int và long (long long)  
Ban đầu, chúng ta có biến **addMaxIntVariable** với giá trị tối đa của kiểu dữ liệu Integer (kiểu dữ liệu mặc định **32 bit = 8 bytes**). Ta có thể thấy, lúc này giá trị của biến **addMaxIntVariable** = **2147483647**. Tiếp đó, ta khai báo một biến **checkMaxIntVariable** có kiểu long long (kiểu dữ liệu có giá trị tối thiểu là **64 bit = 16 bytes**) và gán cho giá trị này bằng giá trị **addMaxIntVariable + 1**. Lúc này, giá trị của biến **checkMaxIntVariable** nên là 2147483648 (2147483647 + 1), nhưng thực tế chúng ta nhận được kết quả là -2147483648.   
\_Lại tiếp tục thực nghiệm, gán giá trị cho biến checkMaxIntVariable các giá trị lần lượt là 2147483648 và LLong\_MAX(9223372036854775807). Tại đây ta có thể thấy giá trị lúc này lớn hơn rất nhiều so với addMaxIntVariable và vẫn hiển thị đúng giá trị ta đã gán.  
\_Lúc này ta có thể đưa ra kết luận: vì biến addMaxIntVariable có giá trị ô nhớ nằm trong vùng 8 Bytes này (tối đa là 2147483647) và được hiển thị với giá trị nhị phân 1111111111111111111111111111111 và có phần bù là 0, khi ta tiến hành + 1, tất cả giá trị trong vùng 8 Bytes sẽ chuyển về 0 và cộng thêm 1. Lúc này, phần cộng thêm 1 sẽ được cộng vào phần bù, do giá trị của biến addMaxIntVariable lúc này chỉ có 8 bytes, và phần 1 đã được thêm vào byte kế tiếp đó (kết quả : (1)0000000000000000000000000000000), ta nhận thấy 0000000000000000000000000000000 chính là bù 1 của 2147483647 và bù 2 = bù 1 + 1, nên lúc này(1)0000000000000000000000000000000 sẽ được biểu diễn dưới dạng bù 2, và kết quả sẽ cho ra là -2147483648.

Bài 5c:

MIN:

Ta có thể thấy đối với 2 số unsigned int(q) và unsigned char(r), thì số nhỏ nhất sẽ là 0, và số nhỏ nhất của int(p) là -2147483648, nên kết quả của phép toán cộng trên sẽ là -2147483648. Tuy nhiên, kiểu dữ liệu của phép toán + trên sẽ là unsigned int, mà theo ta đã biết, unsigned int là kiểu dữ liệu có số min là 0, nên kiểu dữ liệu từ kết quả bài toán là sai. Để có thể thấy rõ lỗi sai, thay vì dùng long x, ta có thể chuyển sang kiểu dữ liệu lần lượt là unsigned và signed int.

MAX:  
Tuy nhiên ở phép cộng, các thứ tự phép toán sẽ làm thay đổi kiểu dữ liệu của kết quả, với phép toán x = p + q + r, ta có thể hiểu gồm 2 kết quả:  
\_ t = p + q và v = t + r  
  
Ta có thể thấy được cho dù đổi vị trí của số int p trong phép toán trên, ta đều sẽ có công thức là một số unsigned int + signed int. Và lúc này, p sẽ bí ép kiểu từ signed int sang unsigned int. Tuy nhiên kết quả vẫn sẽ dẫn đến hiện tượng tràn số và dẫn tính phép tính không còn đúng đắn nữa.