**MÔN LẬP TRÌNH HỆ THỐNG MẠNG**

**LAB 1: MANIPULATING BITS**

# Giới thiệu

Bài lab này tập trung chủ yếu vào kĩ thuật thay đổi giá trị các bit của biến Integer trong ngôn ngữ C. Sinh viên sẽ hoàn thành bài lab thông qua việc giải các câu đố của bài lab, sau đó viết lại đáp án trong một chương trình C/C++ hoàn chỉnh.

**Yêu cầu bài lab:**

Bài lab là một loạt các câu đố, để giải các câu đố, sinh viên cần vận dụng kiến thức trong Chương 2 môn Lập trình Hệ thống mạng. Có thể tham khảo thêm ở sách Computer System: A Programmer’s Perspective (CSAPP).

Trong quá trình giải bài, sinh viên chỉ sử dụng các toán tử **! ˜ & ˆ | + << >>** và không được dùng các phép lập trình cấu trúc (control structures) như loop, conditionals, switch …

Bài lab bao gồm 2 phần chính, phần 2.1 và 2.2 là phần cơ bản, phần 2.3 là phần nâng cao để lấy trọn điểm.

# Nội dung

Mỗi câu đố trong các mục 2.1, 2.2, 2.3 bao gồm 4 nội dung chính:

* **Tên hàm**: Tên của hàm và giá trị trả về chứa nội dung lời giải.
* **Yêu cầu thực thi**: Yêu cầu cần đạt được khi giải câu đố, trong đó bao gồm các phép thử mẫu dùng cho việc đánh giá điểm số (với PT là viết tắt của Phép Thử).
* **Điểm đánh giá**: Mức điểm đánh giá cho mỗi câu đố, mức điểm cao hơn cho biết câu đố đó khó hơn.
* **Max Ops**: Số lượng toán tử tối đa được phép sử dụng trong quá trình lập trình giải câu đố. Nếu số toán tử sử dụng vượt quá số lượng này, và nếu có kết quả đúng, điểm đánh giá sẽ bị chia đôi.

Sinh viên giải bài lab bằng cách viết lại các hàm thực hiện các yêu cầu trong các phần 2.1, 2.2 và 2.3, lưu dưới dạng **MSSV\_cssapLab1.cpp.**

**VD:** bool bitAnd( int x, int y){

…

return … ;

}

## Simple Bit Manipulation

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên hàm** | **Yêu cầu thực thi** | **Điểm** | **Max Ops** |
| boolean bitAnd(x,y)  integer  getByte(x,n)  integer  logicalShift(x,n)  integer  bitCount(x)  bang(x) | Thực thi x & y chỉ dùng toán thử **|** và **~**  PT: bitAnd(3, -9) == (3 & -9)  Lấy byte thứ n từ biến x.  PT: getByte(0x11223344, 1) == 0x33  Dịch phải luận lý một số âm  PT: logicalShift(-1,2) == 0x3fffffff  Đếm số lượng bit mang giá trị 1 của một số 8bit.  PT: unsigned char x = 0xd7  bitCount(x) == 6 && bitCount(0) == 0  Tính giá trị của !x không dùng toán tử !  PT: bang(0) == 1 && bang(-1) == 0 | 1  2  3  4  4 | 8  6  20  40  12 |

## Two’s Compliment Arithmetic

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên hàm** | **Yêu cầu thực thi** | **Điểm** | **Max Ops** |
| integer  tmin()  integer  divpwr2(x,n)  integer  negate(x)  boolean  isPositive(x)  boolean  isLessOrEqual (x,y) | Trả về giá trị số âm nhỏ nhất trong hệ 32bit  PT: tmin() == 0x80000000  Tính giá trị x/2n  PT: divpwr2(-8,1) == -4 && divpwr2(16,2) ==4  Tính giá trị của –x không dùng dấu –  PT: negate(0) == 0 && negate(9) == -9 && negate(-5) == 5  Trả về True nếu x > 0  PT: isPositive(10) && !isPositive(-5) && isPositive(0)  Trả về True nếu x <= y  PT: isLessOrEqual(-1, 4) && !isLessOrEqual(-5, -9) &&  isLessOrEqual(0,0) &&  !isLessOrEqual(5,-9) | 1  2  2  3  4 | 4  15  5  8  24 |

## Advanced Bit Manipulator

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên hàm** | **Yêu cầu thực thi** | **Điểm** | **Max Ops** |
| integer  ilog2(x) | Tính giá trị của Log2(x)  PT: ilog2(1024) == 10 | 4 | 90 |

Tổng điểm : 30/30

Tổng điểm sẽ được chia 3 để tính ra điểm bài Lab.