



Lab 1

BÁO CÁO BÀI THỰC HÀNH SỐ 1 **Data lab: Tính toán Bits**

Data lab: Manipulating Bits

Thực hành Lập trình Hệ thống

Giảng viên hướng dẫn	ThS. Đỗ Thị Hương Lan
Sinh viên thực hiện	Nguyễn Hồ Nhật Khoa (22520677) Lê Quốc Ngô (22520951) Trần Tiến Nhật (22521030)
Mức độ hoàn thành	Hoàn thành
Thời gian thực hiện	08/03/2024 – 11/03/2024
Tự chấm điểm	9/10

2.1 int isSameSign(x, y)

Yêu cầu thực thi:

Kiểm tra x và y có cùng dấu (cùng âm hoặc cùng không âm) hay không. Trả về 1 nếu cùng dấu, 0 nếu không cùng dấu.

- Input: Số nguyên x, y
- PT: `isSameSign(4,10)==1 && isSameSign(1,-5)==0 && isSameSign(0,2)==1`

Code:

```
int isSameSign(int x, int y)
{
    return !((x ^ y) >> 31);
}
```

Giải thích:

- $x \wedge y$: Phép XOR giữa x và y trả về một số mà bit 1 chỉ xuất hiện ở các vị trí mà x và y khác dấu.
- $(x \wedge y) \gg 31$: Dịch phải 31 bit của kết quả phép XOR, lấy bit dấu của kết quả.
- $!((x \wedge y) \gg 31)$: Trả về true nếu bit dấu của x và y giống nhau (cùng dấu), và trả về false nếu bit dấu của chúng khác nhau (khác dấu).

2.2 int is8x(x)

Yêu cầu thực thi:

Kiểm tra 1 số nguyên x có chia hết cho 8 hay không? Trả về 1 nếu chia hết ngược lại trả về 0.

- Input: Số nguyên x không âm
- PT: `is8x(16) == 1 && is8x(3) == 0 && is8x(0) == 1`

Code:

```
int is8x(int x)
{
    return !(x & 7);
}
```

Giải thích:

- $x \& 7$: Phép AND giữa x và 7, số 7 là 111 (3 bits cuối cùng là 1). Nếu tất cả ba bit cuối cùng của x đều là 0, tức là số đó chia hết cho 8.

- $!(x \& 7)$: Nếu tất cả ba bit cuối cùng của x đều là 0, thì kết quả của phép AND là 0 và hàm trả về true. Ngược lại, nếu ít nhất một trong ba bit cuối cùng của x là 1, thì kết quả của phép AND là khác 0 và hàm trả về false.

2.3 int isPositive(x)

Yêu cầu thực thi:

Trả về 1 nếu x dương ($x > 0$)

- Input: Số nguyên x
- PT: $\text{isPositive}(10) == 1 \ \&\& \ \text{isPositive}(-5) == 0 \ \&\& \ \text{isPositive}(0) == 0$

Code:

```
int isPositive(int x)
{
    return !((x >> 31) & !!x);
}
```

Giải thích:

- $x \gg 31$: Trích xuất bit dấu của số nguyên x . Bit dấu là bit cuối cùng sau khi dịch phải 31 bit. Nếu x là số dương, bit dấu sẽ là 0; nếu x là số âm, bit dấu sẽ là 1.
- $!(x \gg 31)$: Trả về 1 (true) nếu bit dấu là 0, tức là x là số dương hoặc 0. Nếu bit dấu là 1, tức là x là số âm, kết quả sẽ là 0 (false).
- $!!x$: Nếu x khác 0, kết quả sẽ là 1; nếu x bằng 0, kết quả sẽ là 0. Nói cách khác, nếu x khác 0, kết quả sẽ là true; nếu x bằng 0, kết quả sẽ là false.
- $!(x \gg 31) \ \& \ !!x$: Kết quả cuối cùng sẽ là 1 (true) nếu x là số dương hoặc 0, và là 0 (false) nếu x là số âm hoặc 0.

2.4 int isLess2n(x,n)

Yêu cầu thực thi:

· Trả về 1 nếu $x < 2^n$

- Input:
- Số x nguyên dương
- Số $n(0 < n < 30)$

PT: $\text{isLess2n}(12,4) == 1 \ \&\& \ \text{isLess2n}(8,3) == 0 \ \&\& \ \text{isLess2n}(15,2) == 0$

Code:

```
int isLess2n(int x, int y)
{
    return !(x & (~(1 << y) + 1));
}
```

Giải thích:

- $1 \ll y$: tính toán giá trị của 2^y bằng cách dịch trái số 1 y lần.
- $\sim(1 \ll y) + 1$: tạo ra một số âm có bit 1 từ bit MSB đến vị trí y và bit 0 ở tất cả các vị trí còn lại.
- $x \& (\sim(1 \ll y) + 1)$: Phép AND này sẽ kiểm tra xem bit thứ y của số x có bằng 0 hay không. Nếu x nhỏ hơn 2^y , thì bit tương ứng của x sẽ bằng 0, và kết quả của phép AND sẽ là 0.
- $!(x \& (\sim(1 \ll y) + 1))$: Toán tử ! được sử dụng để đảo ngược kết quả của phép AND. Nếu bit thứ y của x bằng 0, kết quả sẽ là true, ngược lại kết quả sẽ là false.

Kết quả:

```
Lab1-Nhom7-22521030-22520951-22520677-P2 } ; if ($?) { .\Lab1-Nhom7-22521030-22520951-22520677-P2 }
Your evaluation result:
1.1 bitAnd      Pass.
1.2 negative    Pass.
1.3 getByte     Pass.
1.4 getnbit     Pass.
1.5 mulpw2      Advanced Pass.
2.1 isSameSign  Pass.
2.2 is8x        Pass.
2.3 isPositive  Pass.
2.4 isLess2n    Pass.
--- FINAL RESULT ---
Score: 10.5
Excellent. We found a master in bit-wise operations :D
```