Họ tên: Nguyễn Thị Minh Châu – MSSV: 21520645

Báo cáo LAB 1

1.5 thêm phần nâng cao

```
int mulpw2(int x, int n)
{
    int result;
    PrintBits(x);
    PrintBits(n);
    int i = n >> 31 ; // lấy bit dấu của n để xem n âm hay n dương
    PrintBits(i);
    int x1 = x >> (~n + 1); // x1 tính 2^n với n âm
    int x2 = x << n; // x2 tính 2^n với n dương
    result = (~i & (x2)) | (i & (x1)); // ta dùng cấu trúc MUX 2 to 1
    // để chọn xem nên lấy giá trị x1 hay x2. Nếu i = 0 tức là số dương
    // thì nó sẽ chọn x2, ngược lại nó sẽ chọn x1
    PrintBits(result);
    return result;
}
```

2.1

```
int isSameSign(int x, int y)
{
    int result;
    PrintBits(x);
    PrintBits(y);
    PrintBits((x >> 31)); // dịch phải 31 bit để lấy bit dấu của x
    PrintBits((y >> 31)); // dịch phải 31 bit để lấy bit dấu của y
    PrintBits(!((x >> 31) ^(y >> 31))); // ta XOR hai bit dấu lại
    // rồi lấy giá trị phủ định để được kết quả như đề yêu cầu
    result =  !((x >> 31) ^(y >> 31));
    return result;
}
```

2.2

```c
int is8x(int x)
{
    int result;
    PrintBits(x);
    PrintBits((x ^ 0x111) & 1); // nếu x chia hết cho 8 thì 3 bit cuối
    // của x phải là 0. Ta XOR với 0x111 để ra kết quả như đề yêu cầu
    // rồi AND với '1' để lấy bit bên phải cùng nhất
    result =(x ^ 0x111) & 1;
    return result;

}
```

2.3

```c
int isPositive(int x)
{
    int result;
    PrintBits(x);
    PrintBits((x >> 31));
    PrintBits((!x) ^ !(x >> 31)) ; // lấy bit dấu XOR với nhau.
    // Nếu bit dấu là 0 (số dương) thì kết quả trả về 1, ngược lại trả về 0
    result = ((!x) ^ !(x >> 31)) ;
    return result;
}
```

2.4

```c
int isLess2n(int x, int n)
{
    int result;
    PrintBits(x);
    PrintBits(n);
    PrintBits(x >> n); // dịch x sang phải n lần để lấy cái bit có
    //số thứ tự cao hơn n của x, nếu kết quả = 0 nghĩa là các bit thứ tự
    // cao hơn n = 0 nên suy ra x bé hơn 2^n
    PrintBits((x >> n) ^ 1); // XOR kết quả ở trên với 1
    PrintBits(((x >> n) ^ 1) & 0x1); // AND với 0x1 để lấy bit bên phải cùng duy nhất theo đề bài
    result = (((x >> n) ^ 1) & 0x1 );
    return result;
}
```