Laporan Program OnlineGDB

1.1. Fungsi AND

- Logika (x AND y) dapat diganti dengan (NOT (NOT x) OR (NOT y)) sehingga pada program bahasa c dapat ditulis menggunakan bitwise operator (~ (~ x) | (~ y)). Kode lengkapnya sebagai berikut.

```c
#include <stdio.h>

int bitAND(int x, int y)
{
   int result;

   result = ~((~x) | (~y));

   return result;

}

int main()
{
   int x, y;

   scanf("%d", &x);
   scanf("%d", &y);

   printf("%d", bitAND(x, y));

   return 0;
}
```

1.2. Mendapatkan byte ke-n dari variabel x

- Ukuran dari integer adalah 4 byte (32 bit), Apabila ingin mendapatkan 1 byte (8 bit) pada posisi ke n, Maka bit dari integer harus digeser kekanan sejauh 8 * (n - 1). Contoh:
    - Apabila ingin mendapatkan byte pertama, maka tidak perlu digeser. Karena (n - 1) = 0
    - Apabila ingin mendapatkan byte kedua, maka perlu digeser sejauh 1 byte(8 bit). Karena (n - 1) = 1
    - Apabila ingin mendapatkan byte ketiga, maka perlu digeser sejauh 2 byte(16 bit). Karena (n - 1) = 2
    - Apabila ingin mendapatkan byte keempat, maka perlu digeser sejauh 3 byte(24 bit). Karena (n - 1) = 3

  Pergeseran tiap byte kekanan tersebut diperlukan untuk mendapatkan byte yang diinginkan berada pada posisi paling kanan. Setelah byte berada pada posisi paling kanan, Kemudian dipakai logika AND sebesar 0xff (255) [Panjang 1 byte / 8 bit] untuk mendapatkan nilai desimal dari byte yang paling kanan saja.
  Kode lengkapnya sebagai berikut:

```c
#include <stdio.h>

int getByte(int x, int n)
{
   int result = (x >> (8 * (n - 1))) & 255;

   return result;

}

int main()
{
   int x, n;

   scanf("%d", &x);
   scanf("%d", &n);

   printf("%d", getByte(x, n));

   return 0;
}
```

1.3. Shift Right

- Pada soal diminta untuk menggeser bit variable x kekanan sebanyak n kali tanpa menggunakan bitwise operator >>. Pada bilangan biner, basis yang digunakan adalah 2, Apabila suatu bilangan biner digeser 1 kali kekanan nilai desimal bilangan biner yang baru adalah 1 / 2 dari nilai desimal bilangan biner awal. Maka dengan menggunakan looping untuk tiap bit hingga n kali bisa didapatkan nilai desimal bilangan biner yang baru tanpa harus menggunakan bitwise operator >>.
  Kode lengkapnya sebagai berikut:

```c
#include <stdio.h>

int logicalShift(int x, int n)
{
    int result = x;
    for(int i = 0; i < n; i++){
        result = result / 2;
    }
    return result;
}

int main()
{
    int x, n;

    scanf("%d", &x);
    scanf("%d", &n);

    printf("%d", logicalShift(x, n));

    return 0;
}
```