LAB 1

Title

Learning basic about Decimal to Binary and Binary to Decimal

Theory

Decimal = $b_0 \times 2^0 + b_1 \times 2^1 + b_2 \times 2^2 + ...$

$111001_2 = 1*2^5+1*2^4+1*2^3+0*2^2+0*2^1+1*2^0 = 57_{10}$

Binary to Decimal

Division by 2	Quotient	Remainder	Bit #
13/2	6	1	3
6/2	3	0	2
3/2	1	1	1
½	0	1	0

So $13_{10} = 1101_2$

Source Code

//binary to decimal

```
int binaryToDecimal(long binarynum)
{
int decimalnum = 0, temp = 0, remainder;
while (binarynum!=0)
{
remainder = binarynum % 10;
binarynum = binarynum / 10;
decimalnum = decimalnum + remainder*pow(2,temp);
temp++;
}
return decimalnum;
}
```

//decimal to binary

```
long decimalToBinary(long n)
{
int remainder;
long binary = 0, i = 1;
while(n != 0) {
remainder = n%2;
n = n/2;
binary= binary + (remainder*i);
i = i*10;
}
return binary;
}
```

LAB 2

Title

Addition of two unsigned integer binary number

Source Code:

```
int main()
{
   long int binary1,binary2;//declare 2 binary no
   int i=0,remainder = 0,sum[20];

   printf("Enter any first binary number: ");
   scanf("%ld",&binary1);//binary1 input
   printf("Enter any second binary number: ");
   scanf("%ld",&binary2);//binary2 input

   while(binary1!=0||binary2!=0)
        {
      sum[i++] =  (binary1 %10 + binary2 %10 + remainder ) % 2;//sum=sum+a
      remainder = (binary1 %10 + binary2 %10 + remainder ) / 2;
      binary1 = binary1/10;
      binary2 = binary2/10;
        }

   if(remainder!=0)
      sum[i++] = remainder;
        --i;
   printf("Sum of two binary numbers: ");
   while(i>=0)
      printf("%d",sum[i--]);
        return 0;
}
```

LAB 3

Title

Subtraction of two unsigned integer binary number.

Source Code:

```
#include <math.h>
int main(){
int numa[8]={0},numb[8]={0},diff[8]={0};
int i;
printf("\nEnter first number: ");
for(i=0; i<4; i++){
scanf("%d",&numa[i]);
}
printf("\nEnter second number: ");
for(i=0; i<4; i++){
scanf("%d",&numb[i]);
}
for(i=3; i>= 0; i--){
diff[i] = numa[i] - numb[i];
if(diff[i] < 0){
numa[i-1] = numa[i-1] - 1;
}
diff[i] = fabs(diff[i]%2);
}
printf("\nDifference is: ");
for(i=0; i<4; i++){
printf("%d",diff[i]);
}
return 0;
}
```

LAB 4

Title

Multiplication of two unsigned integer Binary numbers

Theory

```
1011  (this is 11 in decimal)
   x 1110  (this is 14 in decimal)
   ======
     0000  (this is 1011 x 0)
     1011    (this is 1011 x 1, shifted one position to the left)
   1011    (this is 1011 x 1, shifted two positions to the left)
 + 1011      (this is 1011 x 1, shifted three positions to the left)
 =========
 10011010  (this is 154 in decimal)
```

Source code

```
#include<stdio.h>
int binaryAddition(int,int);
int main()
{
   long int binary1,binary2,multiply=0;
   int digit,factor=1;
   printf("Enter first 4 bit  binary number: ");
   scanf("%ld",&binary1);
   printf("Enter second 4 bit binary number: ");
   scanf("%ld",&binary2);

   while(binary2!=0){
       digit =  binary2 %10;

       if(digit ==1)
         {
            binary1=binary1*factor;
            multiply = binaryAddition(binary1,multiply);
```

```
        }
      else
         binary1=binary1*factor;

      binary2 = binary2/10;
      factor = 10;
   }

   printf("Product of two binary numbers: %ld",multiply);

  return 0;
}

int binaryAddition(int binary1,int binary2)
{

   int i=0,remainder = 0,sum[20];
   int binarySum=0;

   while(binary1!=0||binary2!=0){
      sum[i++] =  (binary1 %10 + binary2 %10 + remainder ) % 2;
      remainder = (binary1 %10 + binary2 %10 + remainder ) / 2;
      binary1 = binary1/10;
      binary2 = binary2/10;
   }

   if(remainder!=0)
      sum[i++] = remainder;
   --i;
   while(i>=0)
      binarySum = binarySum*10 + sum[i--];
   return binarySum;
 }
```

LAB 5

Title

Division of two unsigned integer Binary numbers

Source Code

```
#include <math.h>
//binary to decimal
int binaryToDecimal(long binarynum)
{
   int decimalnum = 0, temp = 0, remainder;
   while (binarynum!=0)
   {
      remainder = binarynum % 10;
      binarynum = binarynum / 10;
      decimalnum = decimalnum + remainder*pow(2,temp);
      temp++;
   }
   return decimalnum;
}
//decimal to binary
long decimalToBinary(long n)
{
   int remainder;
   long binary = 0, i = 1;
   while(n != 0) {
      remainder = n%2;
      n = n/2;
      binary= binary + (remainder*i);
      i = i*10;
   }
   return binary;
}
int main()
```

```
{
   int quotient,rem;
   long binarynum1,binarynum2;
   int decimalnum1,decimalnum2;
   //input dividend
   printf("Enter dividend binary number: ");
   scanf("%ld", &binarynum1);
   //input divisor
   printf("Enter divisor binary number: ");
   scanf("%ld", &binarynum2);
   decimalnum1=binaryToDecimal(binarynum1);
   decimalnum2=binaryToDecimal(binarynum2);
   printf("\nEquivalent decimal number is: %d",
binaryToDecimal(binarynum1));//first
   printf("\nEquivalent decimal number is: %d",
binaryToDecimal(binarynum2));//second
   quotient=decimalnum1/decimalnum2;
   rem=decimalnum1%decimalnum2;
   printf("\nEquivalent Quotient binary number is: %d",
decimalToBinary(quotient));
   printf("\nEquivalent Remaining binary number is: %d", decimalToBinary(rem));
return 0;
}
```