Assignment 1

Garvit Shah
U21CS089

Prepare the c program for the followings with separate header file consists of all functions that you are using in the programs:

```
U21CS089.h File -
#include <stdio.h>
#include <math.h>
#include <stdlib.h>

//1. Take a decimal number from the command line and convert it
into a binary number.
void intToBin(float n){
    int binL[64], binR[64], i=0, I=fabsf(n), m=0, p=0;
    float F = fabsf(n)-I;
    while(I>0){
        if(I%2 == 0){
            binL[i] = 0;
        }
        else{
            binL[i]=1;
        }
        I = I/2;
        m+=1;
        i+=1;
    }
    if(n<0){
        printf("-");
    }
    for(int i=m-1; i>=0; i--){
        printf("%d", binL[i]);
    }
    printf(".");
    for(int i =0;i<=4;i++){
        p = 2*(F);
        F = (2*F) - p;
        if(p>1){
            break;
```

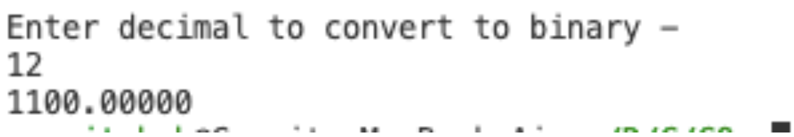
```c
        }
        else{
            printf("%d", p);
        }
    }
    printf("\n");
}

//2. Take a binary number from the command line and convert it into a decimal number.
float binToInt(char bin[120]){
    float dec = 0;
    int m =0, dot = 0;
    while(bin[m]!='.'){
        if(bin[m]=='\0'){
            break;
        }
        m+=1;
    }
    if(bin[m] == '.'){
        dot = '1';
    }
    for(int i = m-1;i>=0;i--){
        if(bin[i] == '1'){
            dec += pow(2,m-1-i);
            // printf("\n%f", dec);
        }
    }
    if(dot == '1'){
        for(int i = m+1;bin[i]!='\0';i++){
            if(bin[i] == '1'){
                dec += pow(2, -(i-m));
            }

        }
    }
    // printf(">> %f", dec);
    return dec;
}
```

```c
//3. Take a decimal number from the command line and display its factorial using recursion.
int fact(int n){
    if(n==0){
        return 1;
    }
    else{
        return(n*fact(n-1));
    }
}

//4. function addition_unsigned_binary_numbers_U22CSXYZ to add two 8-digits unsigned binary given from the command line.
int addition_unsigned_binary_numbers_U22CS089(char b1[120], char b2[120]){
    int c1 = 0, c2=0, m, diff=0;
    while(b1[c1] != '\0'){
        c1+=1;
    }
    while(b2[c2] != '\0'){
        c2+=1;
    }
    m = c1;
    if(c1 > c2){
        m = c1;
        diff = c1-c2;
        for(int i=c2-1; i>=0;i--){
            b2[i+diff] = b2[i];
        }
        for(int i = 0;i<diff;i++){
            b2[i] = '0';
        }
    }
    else{
        m = c2;
        diff = c2-c1;
       for(int i=c1-1; i>=0;i--){
        b1[i+diff] = b1[i];
       }
```

```
   for(int i = 0;i<diff;i++){
    b1[i] = '0';
   }
}
int carr = 0, q = 0;
char add[m+1];
for(int i = m-1;i>=0;i--){
    if(b1[i] == '1' && b2[i] == '1'){
        if(carr == 0){
            add[i+1] = '0';
            carr = 1;
        }
        else{
            add[i+1] = '1';
            carr = 1;
        }
    }
    else if(b1[i]=='1' || b2[i] == '1'){
        if(carr == 1){
            add[i+1] = '0';
            carr = 1;
        }
        else{
            carr = 0;
            add[i+1] = '1';
        }
    }
    else{
        if(carr == 1){
            add[i+1] = '1';
            carr = 0;
        }
        else{
            add[i+1] = '0';
            carr = 0;
        }
    }
}
if(carr == 1){
    add[0] = '1';
    }
    else{
        add[0] = '0';
    }
    printf("\n");
    for(int i =0;i<=m;i++){
        printf("%c", add[i]);
    }
    return 0;
}

//5. function addition_signed_binary_numbers_U22CSXYZ to Add two
8-digits signed binary given from command line.
int addition_signed_binary_numbers_U22CS089(char b1[], char b2[]){
    int d1, d2;
    float sum = 0;
    char s1 = b1[0], s2 = b2[0];
    b1[0] = '0';
    b2[0] = '0';

    d1 = binToInt(b1);
    d2 = binToInt(b2);

    if(s1 == '1'){
        d1 = (-1)*d1;
    }
    if(s2 == '1'){
        d2 = (-1)*d2;
    }
    sum = d1+d2;
    printf("\n%f\n", sum);
    intToBin(sum);
    return 0;
}
```

1. Take a decimal number from the command line and convert it into a binary number.

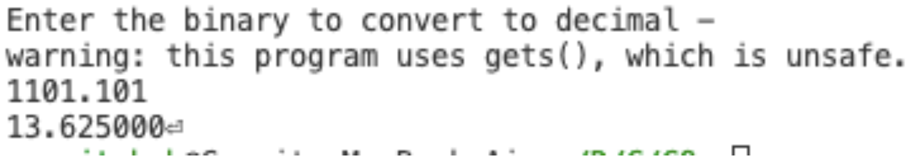
```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "u21cs089.h"

int main(){
    float num;
    int i =0, F, m=0;
    char bin[100], bin1[100], bin2[100];
    char dot;

    printf("\nEnter decimal to convert to binary - \n");
    scanf("%f", &num);
    intToBin(num);
}
```



```
Enter decimal to convert to binary -
12
1100.00000
```

2. Take a binary number from the command line and convert it into a decimal number.

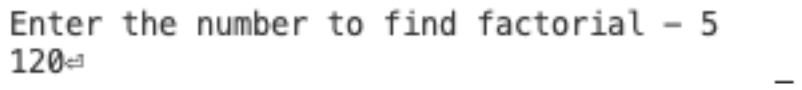
```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "u21cs089.h"

int main(){
    float num;
    int i =0, F, m=0;
    char bin[100], bin1[100], bin2[100];
    char dot;

    printf("\nEnter the binary to convert to decimal - \n");
    gets(bin);
    printf("%f", binToInt(bin));

}
```



```
Enter the binary to convert to decimal -
warning: this program uses gets(), which is unsafe.
1101.101
13.625000⏎
```

3. Take a decimal number from the command line and display its factorial using recursion.

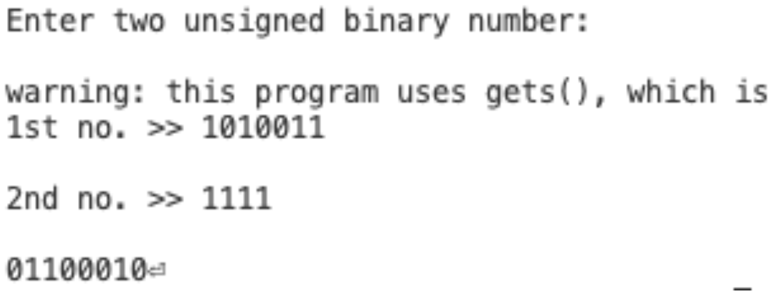
```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "u21cs089.h"

int main(){
    float num;
    int i =0, F, m=0;
    char bin[100], bin1[100], bin2[100];
    char dot;

    printf("\nEnter the number to find factorial - ");
    scanf("%d", &F);
    printf("%d", fact(F));5
}
```



```
Enter the number to find factorial - 5
120⏎
```

4. function addition_unsigned_binary_numbers_U22CSXYZ to add two 8-digits unsigned binary given from the command line.

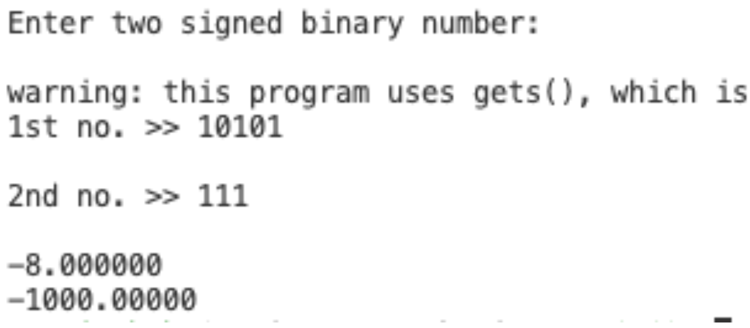
```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "u21cs089.h"

int main(){
    float num;
    int i =0, F, m=0;
    char bin[100], bin1[100], bin2[100];
    char dot;
    printf("\nEnter two unsigned binary number: \n");
    printf("\n1st no. >> ");
    gets(bin1);
    printf("\n2nd no. >> ");
    gets(bin2);
    addition_unsigned_binary_numbers_U22CS089(bin1, bin2);
}
```



```
Enter two unsigned binary number:

warning: this program uses gets(), which is unsafe.
1st no. >> 1010011

2nd no. >> 1111

01100010⏎
```

5. function addition_signed_binary_numbers_U22CSXYZ to Add two 8-digits signed binary given from command line.

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "u21cs089.h"

int main(){
    float num;
    int i =0, F, m=0;
    char bin[100], bin1[100], bin2[100];
    char dot;
    printf("\nEnter two signed binary number: \n");
    printf("\n1st no. >> ");
    gets(bin1);
    printf("\n2nd no. >> ");
    gets(bin2);
    addition_signed_binary_numbers_U22CS089(bin1, bin2);
}
```



```
Enter two signed binary number:

warning: this program uses gets(), which is unsafe.
1st no. >> 10101

2nd no. >> 111

-8.000000
-1000.00000
```