

```

//
//  main.c
//  p1_lib
//
//  Created by Haolin Wang on 3/18/23.
//  ICSI333. System Fundamentals,
//  Spring 2023,
//  TA: Manasa Nannuru
//  Me: Haolin Wang
//  NetID: HW479779
//
#include <stdio.h>
#include "p1_lib.h"

int main(int argc, const char * argv[]) {
    // insert code here...
    printf("Hello, World!\n");
    int i = 0, i2 = 0, countInt = 0, countD=0, inputInt, quotient=0, remainder,
    exp = 0, countIntE = 0, countExp = 0, countManti=0;
    char binaryInt[30], mantiInt[34], exp_Int[10], binaryD[40];
    float inputTop=0.3, inputD = 0, inputManti = 0;

    /* Integer */
    if((int)inputTop-inputTop == 0 && (int)inputTop!=0){
        if(inputTop<0)
            inputInt = inputTop*-1;
        else
            inputInt = (int)inputTop;
        do {
            quotient=inputInt/2;
            remainder=inputInt%2;
            binaryInt[i]=remainder;
            inputInt=quotient;
            i++;
        } while(quotient!=0);

        binaryInt[i]='a';
        while(binaryInt[countInt]!='a'){
            countInt++;
        }
        printf("Input: %f\n", inputTop);
        if(inputTop>=0){
            mantiInt[0]=0;
            printf("Sign: 1\n");
        } else {
            mantiInt[0]=1;
            printf("Sign: 1\n");
        }
        // Generating Mantissa
        i2=1;
        for (i=countInt-1; i>0; i--){

```

```

        mantiInt[i2]=binaryInt[i];
        i2++;
    }
    // filling array with 0s
    while (i2<=23){
        mantiInt[i2]=0;
        i2++;
    }
    i2=0;
    printf("Mantissa (24 bit): ");
    while (i2<=23){
        printf("%d", mantiInt[i2]);
        i2++;
    }
    printf("\n");
    //printf("Digit count is: %d\n", i2);

    // exponent
    exp = countInt-1;
    inputInt = 127+exp;
    do {
        quotient=inputInt/2;
        remainder=inputInt%2;
        exp_Int[i]=remainder;
        inputInt=quotient;
        i++;
    } while(quotient!=0);
    exp_Int[i]='a';
    while(exp_Int[countIntE]!='a'){
        countIntE++;
    }
    while (countIntE<8) {
        exp_Int[countIntE]=0;
        countInt=countInt+1;
        exp_Int[countIntE]='a';
    }
    printf("Exponent: ");
    for (i=countIntE-1;i>=0;i--) {
        printf("%d", exp_Int[i]);fflush(stdout);
    }
    printf("\n");fflush(stdout);
}/* End of Integer */

/* 0.[non-zero] */
else if ((int)inputTop - inputTop != 0 && (int)inputTop == 0){
    if(inputTop<0)
        inputD = inputTop * -1;
    else
        inputD = inputTop;
    inputManti = inputD;
    char binaryD[40];

```

```

i=0;
printf("Input is %f\n", inputTop);
while (inputD-(int)inputD != 0 && i<55){
    inputD = inputD * 2;
    binaryD[i]=(int)inputD;
    if(inputD>1){
        inputD = inputD-(int)inputD;
    }
    i++;
}
binaryD[35]='a';
countD=0;
while(binaryD[countD]!='a'){
    countD++;
}

i=-1;
int countDot=0;
while(binaryD[i]!=1){
    countDot++;
    i++;
}
i=countDot;
countManti=0;
if(inputTop>=0){
    binaryD[countDot-1]=0;
    printf("Sign: 0\n");
} else {
    binaryD[countDot-1]=1;
    printf("Sign: 1\n");
}

// mantissa
printf("Mantissa (24 bit): ");
for (i=countDot-1;i<=22+countDot;i++){
    printf("%d", binaryD[i]);fflush(stdout);
    countManti++;
}
printf("\n");fflush(stdout);
//printf("Mantissa bits: %d\n", countManti);

// exponent
int exp = 0;
exp = countDot;
inputInt = 127-countDot;
int countDE = 0; i = 0;
char exp_D[10];
do{
    quotient=inputInt/2;
    remainder=inputInt%2;
    exp_D[i]=remainder;
    inputInt=quotient;

```

```

        i++;
    } while(quotient!=0);
    exp_D[i]='a';
    while(exp_D[countDE]!='a'){
        countDE++;
    }
    while(countDE<8){
        exp_D[countDE]=0;
        countDE=countDE+1;
        exp_D[countDE]='a';
    }
    printf("Exponent: ");
    for (i=countDE-1;i>=0;i--){
        printf("%d", exp_D[i]);fflush(stdout);
    }
    printf("\n");
} /* End of 0.[non-zero] */
else if (inputTop == 0){
    printf("Everything is 0\n");
}
/* [non-zero].[non-zero]*/
else {
    i=0;
    printf("Input: %f\n", inputTop);
    if (inputTop<0)
        inputInt = inputTop*-1;
    else
        inputInt = (int)inputTop;
    do{
        quotient = inputInt/2;
        remainder = inputInt%2;
        binaryInt[i] = remainder;
        inputInt = quotient;
        i++;
    }while (quotient != 0);
    binaryInt[i]='a';
    while(binaryInt[countInt]!='a'){
        countInt++;
    }
    char binary[60]; int b_i=0;
    for (i=countInt-1;i>=0;i--){
        binary[b_i]=binaryInt[i];
        b_i++;
    }
    // converting post decimal part
    inputD=inputTop-(int)inputTop;
    if(inputTop<0)
        inputD=(inputTop*-1)-((int)inputTop*-1);
    else
        inputD=inputTop-(int)inputTop;
    inputManti=inputD;

```

```

char binaryD[40];
i=0;
while (inputD-(int)inputD != 0 && i<50){
    inputD = inputD*2;
    binaryD[i]=(int)inputD;
    if(inputD>1){
        inputD=inputD-(int)inputD;
    }
    i++;
}
binaryD[39]='a';
countD=0;
while(binaryD[countD]!='a'){
    countD++;
}
// combining 2 arrays
for(i=0;i<35;i++){
    binary[b_i]=binaryD[i];
    b_i++;
}

// mantissa
i2=0;
char mantiInt[34];
for(i=countInt-2; i>= 0; i--){
    mantiInt[i2]=binaryInt[i];
    i2++;
}
char manti[34];
i=0;
if(inputTop>=0){
    manti[0]=0;
    printf("Sign: 1\n");
} else {
    manti[0]=1;
    printf("Sign: 0\n");
}
//mantissa printing
printf("Mantissa (24 bit): ");
for (i=0; i<=23; i++){
    manti[i+1]=binary[i+1];
    printf("%d", manti[i]);fflush(stdout);
}
printf("\n");
printf("Number of mantissa digit is: %d", i);
// exponent
int exp = 0, exp_i;
exp = countInt-1;
inputInt=127+exp;
int countIntE=0; i=0;
char exp_Int[8];

```

```

int countExp = 0;
do {
    quotient=inputInt/2;
    remainder=inputInt%2;
    exp_Int[i]=remainder;
    inputInt=quotient;
    i++;
}while (quotient!=0);
exp_Int[i]='a';
while(exp_Int[countIntE]!='a'){
    countIntE++;
}
while (countIntE<8){
    exp_Int[countIntE]=0;
    countInt=countInt+1;
    exp_Int[countIntE]='a';
}
printf("\n");
printf("Exponent: ");
for (i=countIntE-1;i>=0;i--){
    printf("%d", exp_Int[i]); fflush(stdout);
}
printf("\n");
}

return 0;
}

```