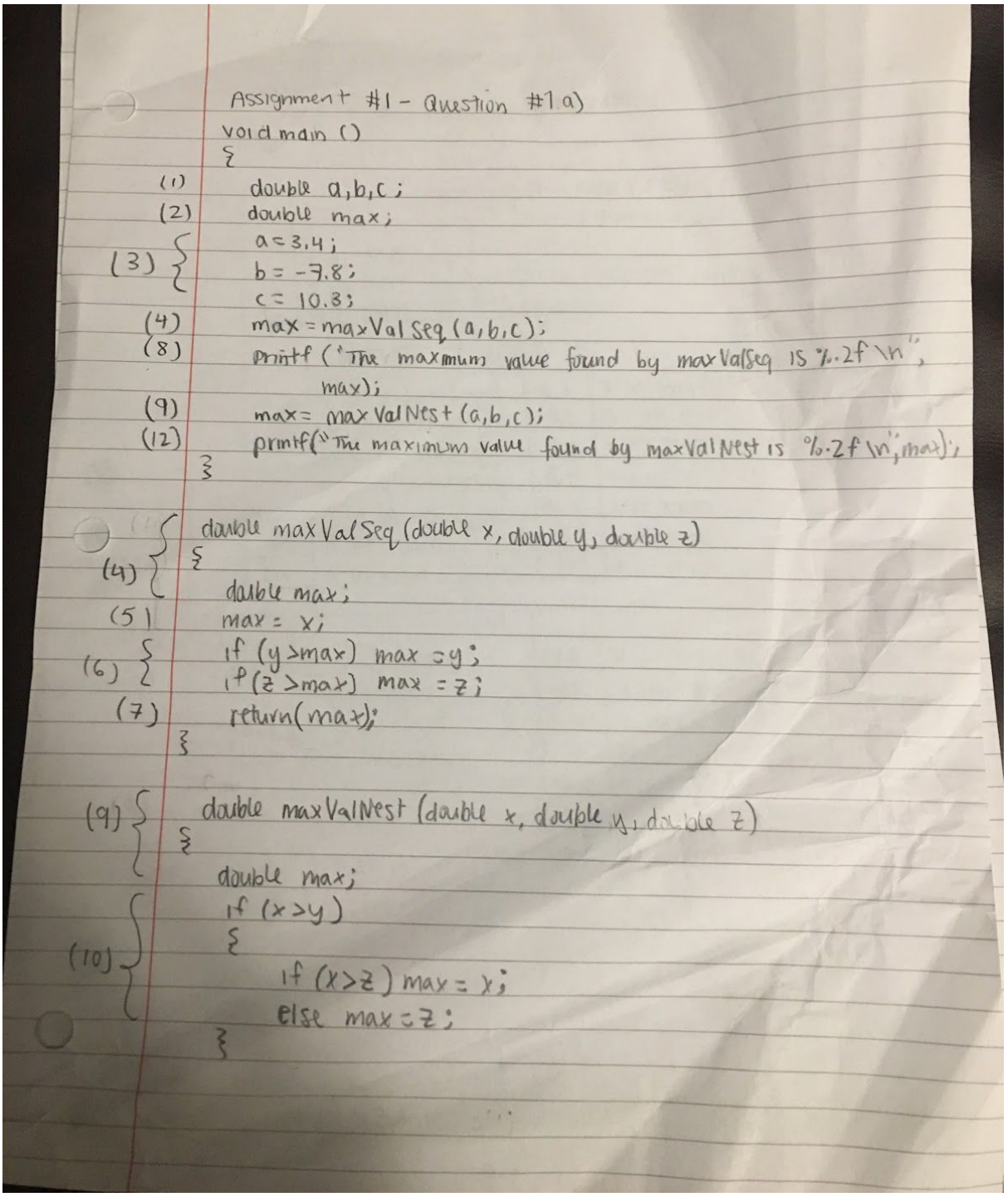
GNG 1106 Assignment #3

Question 1

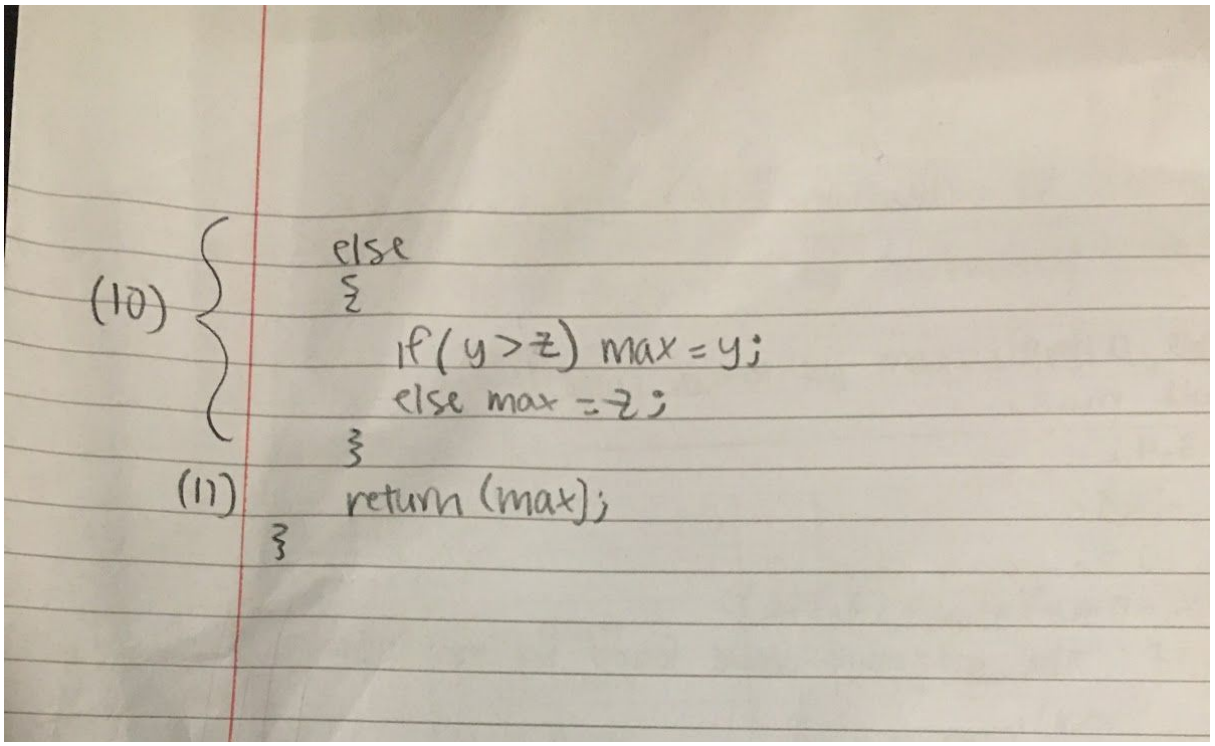
a)



```
Assignment #1 - Question #1 a)
void main ()
{
(1)   double a,b,c;
(2)   double max;
(3)   a=3.4;
      b=-7.8;
      c=10.3;
(4)   max=maxValSeq(a,b,c);
(8)   printf("The maximum value found by maxValSeq is %.2f\n",
          max);
(9)   max=maxValNest(a,b,c);
(12)  printf("The maximum value found by maxValNest is %.2f\n",max);
}

(4)  double maxValSeq(double x, double y, double z)
     {
(5)    double max;
       max=x;
(6)    if(y>max) max=y;
       if(z>max) max=z;
(7)    return(max);
     }

(9)  double maxValNest(double x, double y, double z)
     {
       double max;
(10)   if(x>y)
       {
          if(x>z) max=x;
          else max=z;
       }
```


(10)
else
{
if(y>z) max=y;
else max=z;
}
(11)
return (max);
}

Screen/Console

(8) The maximum value found by maxValSeq is 10.30

(12) The maximum value found by maxValNest is 10.30

(1)

?	c in main
?	b in main
?	a in main

(2)

?	max in main
?	c in main
?	b in main
?	a in main

(3)

?	max in main
10.3	c in main
-7.8	b in main
3.4	a in main

(4)

10.3	z in maxValSeq
-7.8	y in maxValSeq
3.4	x in maxValSeq
?	max in maxValSeq
10.3	c in main
-7.8	b in main
3.4	a in main

(5)

10.3	z in maxValSeq
-7.8	y in maxValSeq
3.4	x in maxValSeq
3.4	max in maxValSeq
10.3	c in main
-7.8	b in main
3.4	a in main

(6)

10.3	z in maxValSeq
-7.8	y in maxValSeq
3.4	x in maxValSeq
10.3	max in maxValSeq
10.3	c in main
-7.8	b in main
3.4	a in main

(9)

10.3	z in maxValNest
-7.8	y in maxValNest
3.4	x in maxValNest
?	max in maxValNest
10.3	c in main
-7.8	b in main
3.4	a in main

(7)

?	max in main
10.3	c in main
-7.8	b in main
3.4	a in main

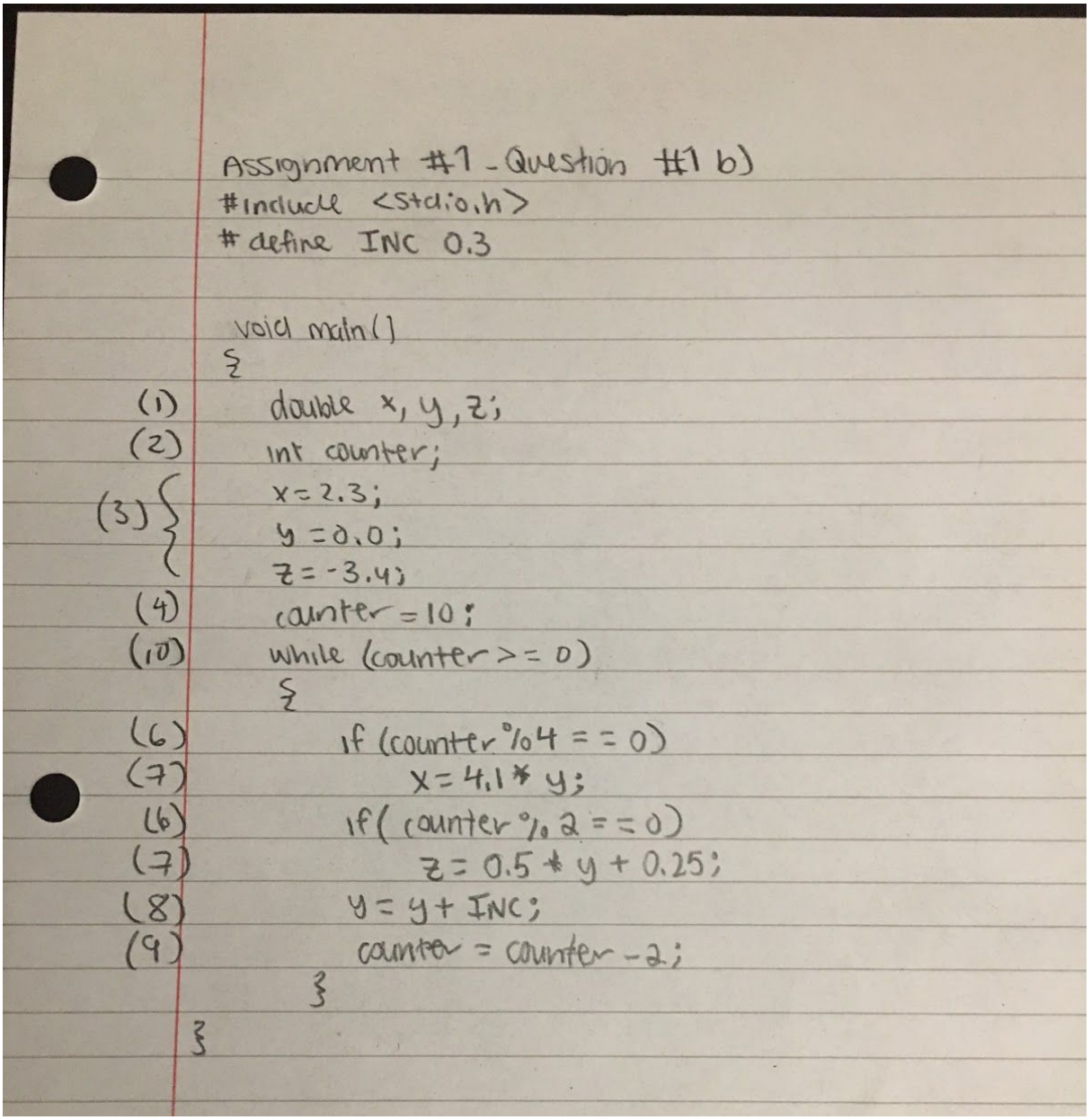
(10)

10.3	z in maxValNest
-7.8	y in maxValNest
3.4	x in maxValNest
10.3	max in maxValNest
10.3	c in main
-7.8	b in main
3.4	a in main

(11)

10.3	c in main
-7.8	b in main
3.4	a in main

b)



```
Assignment #1 - Question #1 b)
#include <stdio.h>
#define INC 0.3

void main()
{
(1)     double x, y, z;
(2)     int counter;
(3)     x = 2.3;
        y = 0.0;
        z = -3.4;
(4)     counter = 10;
(10)    while (counter >= 0)
        {
(6)         if (counter % 4 == 0)
(7)             x = 4.1 * y;
(6)         if( counter % 2 == 0)
(7)             z = 0.5 * y + 0.25;
(8)         y = y + INC;
(9)         counter = counter - 2;
        }
}
```

(1)

?	z in main
?	y in main
?	x in main

(2)

?	counter in main
?	z in main
?	y in main
?	x in main

(3)

?	counter in main
-3.4	z in main
0.0	y in main
2.3	x in main

(4)

10	counter in main
-3.4	z in main
0.0	y in main
2.3	x in main

⇒ (10) counter >= 0

do loop while counter is >=0

(6) counter %4 == 0

(7) x = 4.1 * y

(6) counter %2 == 0

[counter = 10 goes to counter %2]

(7) z = 0.5 * y + 0.25

(8) y = y + INC

(9) counter = counter - 2

[1 LOOP EXAMPLE]

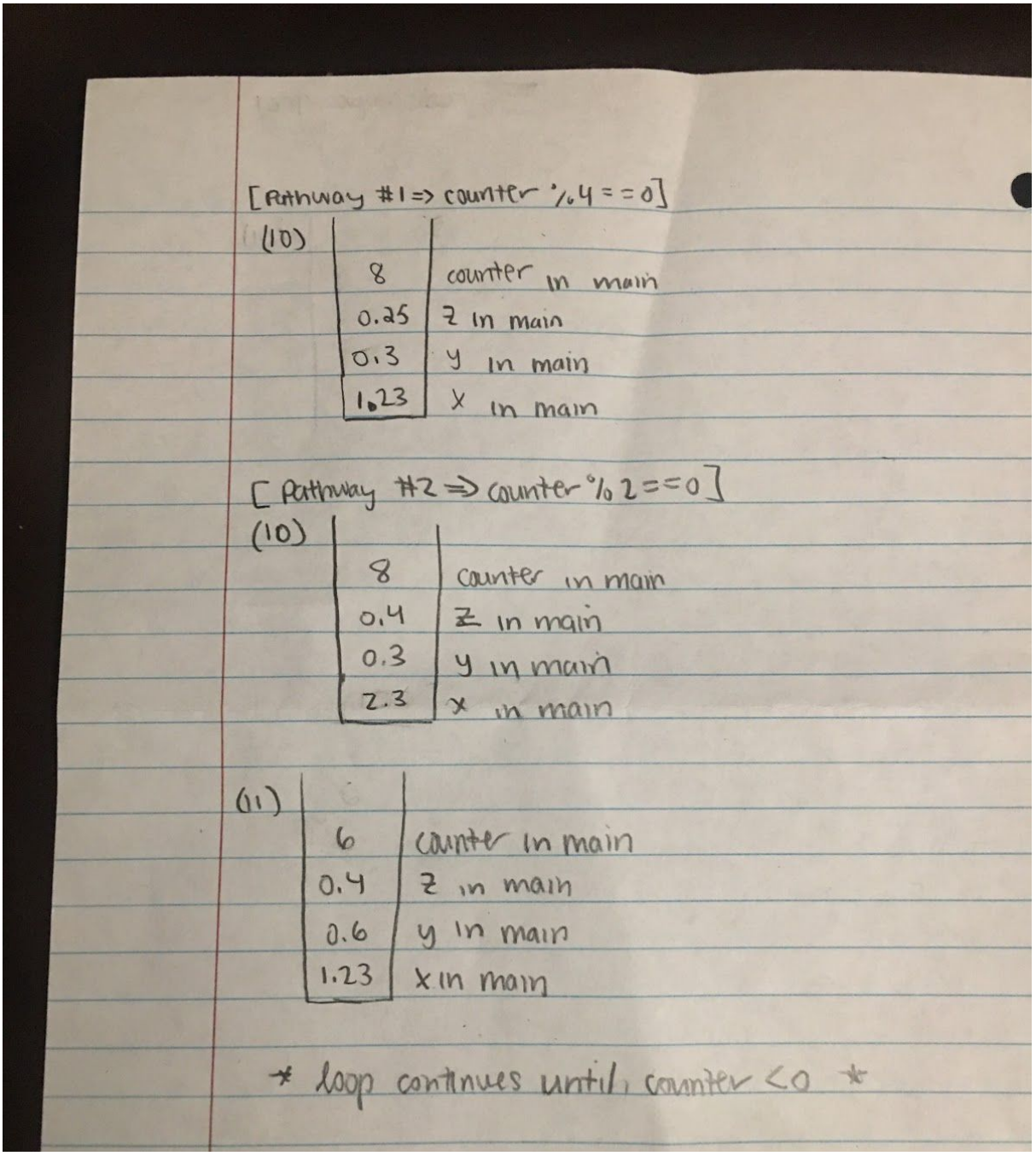
(7)

10	counter in main
0.25	z in main
0	y in main
2.3	x in main

(8/9)

8	counter in main
0.25	z in main
0.3	y in main
2.3	x in main

→ loop until counter < 0



Question 2

The console will print out

"case 1
case 2
default
default
default"

This is because the functions begins with a = 1 which allows it to follow the pathway for case 1. This case will print out "case 1" and then complete the a++ thus making a = 2. It will break from the loop and then repeat starting from a =2. This allows it to follow the pathway for case

2. In case 2, it will print “case 2” and then do a = (2) +2 thus making a = 4. It will break and then repeat starting with a=4. Because a does not equal to 3, it will skip case 3 and move to the default pathway. The default pathway will print “default” and then do a = (4) + 4 thus making a =8. The loop will break and repeat starting with a = 8. Again it will follow the default pathway since a does not equal 1, 2, or 3. It will print “default” again and then do a=(8)+4 thus making a=12. It will repeat this once more under the default pathway thus printing “default” again, but since a= (12)+4 would make a = 16, the loop would stop because the condition of a<13 would not be fulfilled.

Question 3

```
/* program that calculates the Nth prime value */
#include <stdio.h>

#define TRUE 1 // defines boolean values 1 and 0 as true and false
#define FALSE 0

int getNum(int N)
{
    int flag; // declares variables
    do
    {
        printf("Please enter an integer value greater than 10\n");
        // prompts user for value
        scanf("%d", &N);
        // value inputted stored in N

        if(N<10) // loop where pathway followed if N<10
            {
                printf("Value must be greater than 10\n");
                // tells user that input was invalid and loop will repeat
                flag = 0;
            }
        else
            {
                flag = 1;
            }
    }
    while(flag == 0); // repeat loop while condition is true
    return N;
}

int determineIfPrime()
{
    int N, i, flag = 0;
    for(i=2; i<N; i++)
    {
        if (N%i == 0)
        {
            //flag=0;
```

```
            flag = flag+1;
            return flag;
        }
        else
        {
            flag = 1;
        }
     }
}

int main()
{
    int N, i, F, prime = 2; // declares variables
    N = getNum(N);

    for (i=2; i<N; i++) //
    {
        if(determineIfPrime(i) == 1) //
        prime = i;
    }
    printf("The largest prime number that is smaller than %d is %d\n", N,
prime);
    return 0;
}
```

Question 4

```
/* program that calculates the Nth fibonacci value */

#include <stdio.h>

#define TRUE 1 // defines boolean values 1 and 0 as true and false
#define FALSE 0

int calculateFibonacci(int num)
// function used to calculate fibonacci with parameters including num
{
    if (num <= 1) // if num is <=1, it will follow this pathway
             {
            return num; // the value inputted will be the fibonacci value if
it is or under 1
        }
    else // follow this pathway if num is not <=1
    {
        return calculateFibonacci(num - 1) + calculateFibonacci(num - 2);
        // returns the value calculated back to the main function. it adds
the Nth fibonacci number with the one prior to it
    }
}
```

```
int main()
{
      int num, flag; //declares variables
    do
    {
        printf("Please enter a value which you would like the fibonacci
number of\n");
        //prompts user for number which they would like the fibonacci number
of
        scanf("%d", &num);
        // receives value from user and stores it in variable called num

        if(num<0) // follow this pathway if num<0
            {
                printf("Value must be greater than 0\n"); // tells user that
input is invalid
                flag = 0; // continues loop since flag is false
            }
        else // follow this pathway if num>0
            {
                flag = 1; // breaks loop since flag is true
            }
    }
    while(flag == 0); // loop will continue as long as the the condition is
true
      printf("The %dth fibonacci number is %d\n", num,
calculateFibonacci(num));
    // prints out final statement while inputting calculated values
      return 0; //terminates the function
}
```

Question 5

```
/* program that calculates values of the hyperbolic cosine */

#include <stdio.h>

#define FALSE 0 // defines boolean values 1 and 0 as true and false
#define TRUE 1

double getNumTerms(double n)
// function called getNumTerms with parameters including n
{
    double flag; // declares variable
    do
    {
        printf("How many terms to use: ");
        // prompts the user for the amount of terms used
        scanf("%lf", &n);
        // receives the input and stores it into the variable n
```

```
        if (n < 0) // loop where the if pathway will be followed if the
condition is true
        {
            printf("Value must be greater than 0\n");
            // asks user to input a positive number
            flag = 0; // will repeat loop since input is invalid making flag
false
        }
        else // will follow this pathway if n>0
        {
            flag = 1; // will break loop since input is valid making flag
true
        }
    }
    while (flag == 0); // will continue loop while flag is false

    return n; // returns n to main function
}

double pow(double base, double power)
// function called pow to calculate exponents. parameters include base and
power
{
    double result, i; // declares variables
    result = 1; // assigns value of 1 to result
    if (power == 0) // follows this pathway if power is 0
    {
        return 1;
    }

    for (i = 0; i < power; i++)
    // loop will continue starting from i=o, i<0, and i will increase by an
increment of 1 each time
    {
        result *= base; //multiplies base to result and then assigns that
value to result
    }
    return result; // returns result to the main function
}

double coshhyper(double x, double n)
// function called coshhyper. parameters include x and n
{
    double coshx, i; // declares variables
    for (i = 0; i <= n; i++) //condition for i each time the loop is
completed
    {
        coshx += (pow(x, 2 * i)) / (calculateFactorial(2 * i));
        // calls the functions pow and calculateFactorial, calculates their
answers, and stores it into coshx
    }
```

```
    return coshx; // returns coshx to main function
}

int calculateFactorial(double number)
// function called calculateFactorial to calculate factorials. parameters
include number
{
    double i, factorial = 1; // declares variables and assigns 1 to factorial
    for (i = 1; i <= number; i++) // condition for i each time the loop is
completed
    {
        factorial *= i; // multiplies i and factorial and then stores it into
factorial
    }
    return factorial; // returns factorial to main function
}

void main()
{
    double x, n, coshx, result, repeat = 0; // declares variables and
ititializes repeat flag as false
    char answer; // declares answer as a character
    do
    {
        printf("Please enter a real value for x: ");
        // prompts the user for the value of x
        scanf("%lf", &x);
        // receives the value and stores that input in the variable x

        n = getNumTerms(n); // calls the function getNumTerms
        coshx = coshhyper(x, n); // calls the function coshhyper
        printf("cosh(%lf) is %lf\n", x, coshx); // prints out final statement
of what coshx is
        do
        {
            printf("Do you wish to quit (y/n): "); // asks the user if they
want to quit
            scanf(" %c", &answer); // receives input and stores into answer
            if (answer == 'n') // loop which will repeat entire program if
user inputs 'n'
            {
                repeat = 1;
                break;
            }
            else if (answer == 'y') // loop which will terminate program if
user inputs 'y'
            {
                repeat = 0;
                printf("Program terminated\n");
                break;
            }
```

```
            else { // will ask the user if they want to quit again
                repeat = 0;

        }
    }
    while (answer != 'y' || answer != 'n');
    // repeat asking user if they want to quit if this condition is true
(answer is not 'n' or 'y')
}
  while (repeat == 1); // repeat entire program if this condition is true
}
```

```
Please enter a real value for x: 0
How many terms to use: 3
cosh(0.000000) is 1.000000
Do you wish to quit (y/n): a
Do you wish to quit (y/n): b
Do you wish to quit (y/n): c
Do you wish to quit (y/n): A
Do you wish to quit (y/n): Z
Do you wish to quit (y/n): 3
Do you wish to quit (y/n): n
Please enter a real value for x: 1
How many terms to use: 1
cosh(1.000000) is 1.500000
Do you wish to quit (y/n): n
Please enter a real value for x: -1.5
How many terms to use: 7
cosh(-1.500000) is 2.352409
Do you wish to quit (y/n): n
Please enter a real value for x: 35
How many terms to use: 10
cosh(35.000000) is -3546383456118743498752.000000
Do you wish to quit (y/n): n
Please enter a real value for x: -15.3
How many terms to use: 50
cosh(-15.300000) is -13774391927107422765386700437519
Do you wish to quit (y/n): n
Please enter a real value for x: 5.6e-3
How many terms to use: 30
cosh(0.005600) is 1.000016
Do you wish to quit (y/n): y
Program terminated
Alyssas-MacBook-Air-4:Downloads alyssawang$
```