Information and Computer Science Department
ICS 103: Computer Programming in C
Midterm Exam, Term 181
Wednesday, October 31, 2018
Duration: 120 minutes

Name: _____ ID: _____

Instructor and Section: Select one

Instructor	Section		
Dr. El-Sayed El-Alfy	[ ] **01** (UT 8 – 8:50)	[ ] **02** (UT 9 – 9:50)	[ ] **03** (UT 10 – 10:50)
Dr. Hamzah Luqman	[ ] **04** (UT 11 – 11:50)	[ ] **05** (UT 13 – 13:50)	[ ] **06** (UT 14 – 14:50)
Dr. Mohamed Balah	[ ] **07** (UT 15:20 – 16:10)	[ ] **08** (UT 16:20 – 17:10)	
Dr. Tareq El-Bassuny	[ ] **09** (UT 9 – 9:50)	[ ] **10** (MW 9 – 9:50)	[ ] **11** (MW 10 – 10:50)
Dr. Samer Arafat	[ ] **12** (MW 11 – 11:50)	[ ] **13** (MW 13 – 13:50)	
Dr. Yahya Osais	[ ] **14** (MW 14 – 14:50)		
Dr. Kamal C	[ ] **15** (MW 15:20 – 16:10)	[ ] **16** (MW 16:20 – 17:10)	[ ] **21** (UT 13 – 13:50)
Dr. Uthman Baroudi	[ ] **17** (UT 8 – 8:50)	[ ] **18** (MW 9 – 9:50)	
Dr. Ayman Hroub	[ ] **19** (UT 10 – 10:50)	[ ] **20** (UT 11 – 11:50)	

Instructions:
1. Answer all questions. Make sure your answers are **clear** and **readable**.
2. The exam is closed book and closed notes. No calculators or any helping aides are allowed. Make sure to turn off your mobile phone and keep it in your pocket.
3. If there is no space on the front of the page, use the back of the page. Indicate this clearly.

Question	Maximum Points	Earned Points	Remarks
1	33		
2	6		
3	7		
4	20		
5	17		
6	17		
Total	100		

Question 1 : 33 points

What is the output of the following code fragments?

Code fragment	Output

`// 2 pts`
<pre>#include<stdio.h>
int main() {
  int x = 5;
  if(x=5)
  {
    if(x=4)
      printf("Hello");
    else
      printf("Bye");
  }
  printf("Hi");
  return 0;
}</pre>
	HelloHi

`// 2 pts`
<pre>#include <stdio.h>
 int main() {
 int score = 7;
if(score > 15)
   printf("A");
 else if (score)
   printf("B");
 else
   printf("G") ;
if(score <= 5)
   printf("C");
 else if (score>10)
   printf("D");
 printf("E");
return 0 ;
}</pre>
	BE

`// 3 pts`
<pre>#include <stdio.h>
int main() {
  int n = 0;
  while (n < 6) {
    n++;
    if (n % 2 == 1)
      printf("%d\n", n);
  }
  return 0;
}</pre>
	1
	3
	5

``c
                                       // 3 pts
#include <stdio.h>
int main() {
    int i = 1, j = 2;
    for(j++ ; i+j<12; i++){
       printf("%d ",i+j);
       j=j+i;
    }
   return 0;
}
``
	4 6 9

``c
                                      // 3 pts
#include <stdio.h>
int function1(int x)
{
  x = 2;
  printf("x = %d\n",x);
  return(x+1);
}
int main()
{
  int x = 4, y;
  y = function1(x);
  printf("x = %d\n",x);
  printf("y = %d\n",y);
  return 0;
}
``
	x = 2
	x = 4
	y = 3

``c
                                     // 4 pts
#include <stdio.h>
void function2()
{
  printf("two\n");
}
void function1()
{
  function2();
  printf("one\n");
}
void function3()
{
  printf("three\n");
  function2();
}
int main()
{
  function1();
  function3();
  return 0;
}
``
	two
	one
	three
	two

Code	Output

<pre>   //   2 pts
#include<stdio.h>
int main()
{
  int i;
  for(i = 0;i<=3;i++) ;
  printf("%d", i);
     return 0;
}</pre>
	4

<pre>   // 4 pts
#include <stdio.h>
int main()
{
   int m,j,rows=4;
      for(m=1;m<=rows;m++)
   {
      for(j=m;j>=1;j--)
        printf("%d",j);
      printf("\n");
   }
   return 0;
}</pre>
	1
	21
	321
	4321

<pre>   // 4 pts
#include <stdio.h>
int main() {
   int i;
   for(i = 10; i > 0; i-=6){
     printf("%d   ", i+2);
     printf("%d\n", --i);
   }
 return 0;
 }</pre>
	12 9
	5 2

Code	Output

``c
                      // 2 pts
#include <stdio.h>
int main(){
  int item = 2;
  switch (item) {
   case 1: printf("Pen\n") ;
           break;
   case 2:
   case 3: printf("Pencil\n") ;
   case 4:
   case 5:
   case 6: printf("Book\n");
           break;
 default: printf("Notebook\n") ;
}
return 0;
}
``
	Pencil
	Book

``c
                      // 2 pts
#include <stdio.h>
int main(){
int i = 2;
for (i++; i == 3; i = 2)
     printf("%d ", i++);
     printf("%d ", i);
return 0;
}
``
	3 2

``c
                      // 2 pts
#include <stdio.h>
int main(){
 int x;
 scanf("%d", &x);
 do{
    printf("%d\n", x);
    x--;
 } while (!x == 0);
 printf("%d\n", x);
 return 0;
}
 	
// Assume that the value of x entered
by the user is 1
``
	1
	0

Question 2: 6 points

Find the values of the following C expressions.

C expression	Value
5>6-2	1
457/100%10	4
15/6*6.0	12.0
11+1/2.0	11.5
(int)13.4/2.0	6.5
5==5-3	0

Question 3: 7 points

Convert the following *switch* statement to multi-way *if* statement.

```
switch (number)
{
case 10:
    printf("Ten");
    break;
case 20:
    printf("Twenty");
case 30:
case 40:
    printf("Forty");
    break;
    printf("Number 40");
case 50:
    printf("Fifty");
    break;
default:
     printf("Unknown number");
}
```

```
if(number == 10)
    printf("Ten");
else if(number == 20)
{
   printf("Twenty");
   printf("Forty");
}
else if(number == 30 || number== 40)
    printf("Forty");
else if (number == 50)
    printf("Fifty");
else
    printf("Unknown number");
```

Question 4 : 20 points

Write a C program that determines and displays the financial state of a bank client based on his income, credit history (Good, Bad, or Unknown), and debt history (High or Low). The figure below shows the decision tree used to determine the financial state of a bank client. Credit history is inserted by the user as **G** (Good), **B** (Bad), or **U** (Unknown). Debt history is inserted by the user as **H** (High) or **L** (Low).

The following examples demonstrate how the financial state is determined:

- **Example 1:** If the client income is 6000$ then his financial state is High risk.
- **Example 2:** If the income is 29000$ and his credit history is Unknown(U), then we will check his debt history if it is High(H) then his financial state will be High Risk else it will be Moderate Risk.
- **Example 3:** If the income is 75000$ and his credit history is Good(G), then his financial state is Low risk.

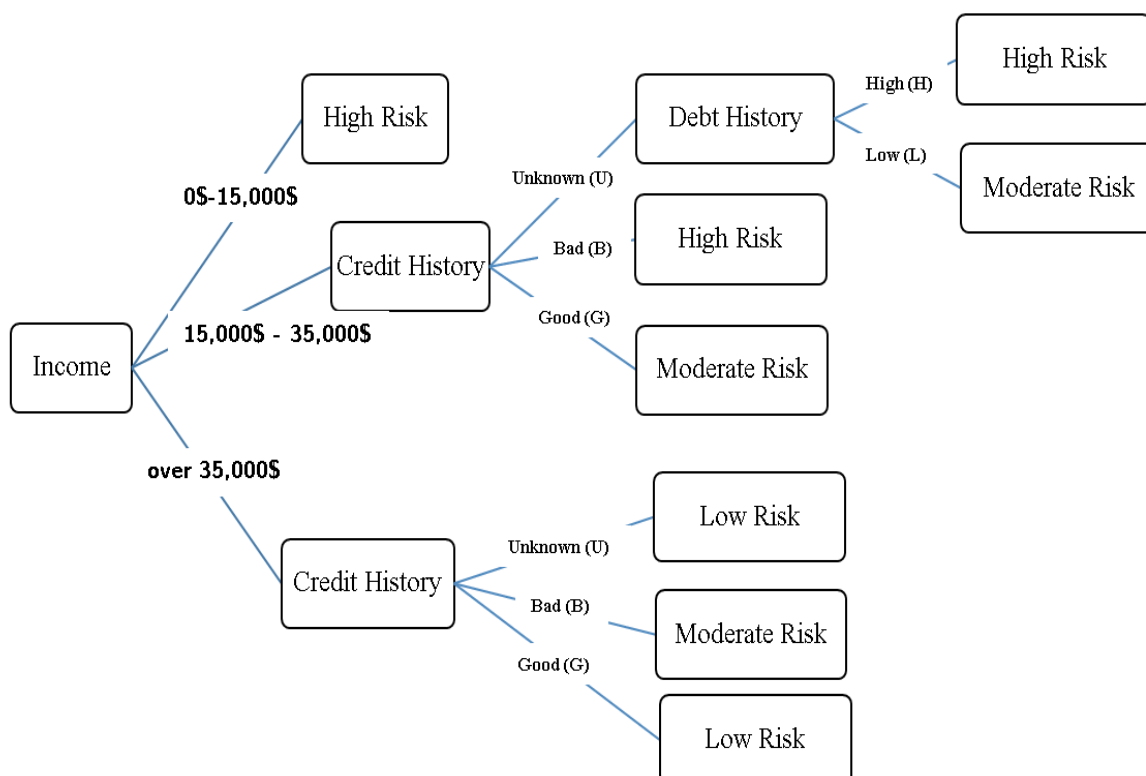**Sample run 1**

```
Insert income: 6000
State: High Risk
```

Sample run 2

```
Insert income: 29000
Insert credit history (U, B, or G): U
Insert debt (H or L): L
State: Moderate  Risk
```

Sample run 3

```
Insert income: 75000
Insert credit history (U, B, or G): G
State: Low Risk
```

The following decision tree will help in the understanding of how to determine the financial state of a bank client:

Solution key:

```c
#include <stdio.h>
int main()
{
  double income;
  char credit, debt;
  printf("insert income: ");
  scanf("%lf", &income);

  if (income >=0 && income <= 15000)
     printf("State: High Risk\n");
  else if (income >15000 && income <= 35000)
  {
     printf("Insert credit history (U, B, or G): ");
     scanf(" %c", &credit);
     switch (credit)         {
          case 'U':
               printf("Insert debt history (H or L): ");
               scanf(" %c", &debt);
               if (debt == 'H')
                    printf("Financial State: High Risk\n");
               else if(debt == 'L')
                    printf("State: Moderate Risk\n");
               break;
          case 'B':
               printf("State: Moderate Risk\n");
               break;
          case 'G':
               printf("State: Moderate Risk\n");
     }

  }
  else
  {
     printf("\nInsert credit history (U, B, or G): ");
     scanf(" %c", &credit);
     if (credit == 'U' || credit =='G')
         printf("State: Low Risk");
     else if(credit == 'B')
         printf("State: Moderate Risk");

  }
  return 0;
}
```

Question 5 : 17 points

Write a complete C program to compute the following power series expansion based on the values of n and x input by the user:

power = $1 + x + x^2 + x^3 ... + x^n$
The condition on x and n are: $-1 < x < +1$ and $n > 0$

Your program must contain 2 functions: **powerSeries** and the **main** function.
powerSeries function receives the values of x and n and returns the value of the power series shown above.
Your program will read from the user the values of x and n. If x and n are within their valid ranges which are $-1 < x < +1$ and $n > 0$, the program computes and displays the value of the power series using the **powerSeries** function. If not, it must print one of the following messages and terminates:
 "n is out of range" or "x is out of range".

Note: No printf or scanf statements inside **powerSeries** function. All reading and printing must be done in the main function.
You must put the definition of **powerSeries** function after the main function

Sample run1

```
Enter x and n : 4.3    45
x is out of range
```

Sample run2

```
Enter x and n : 0.55   -6
n is out of range
```

Sample run3

```
Enter x and n :0.55   65
power series for x=0.55 and n=65 is 2.22
```

Solution:

```c
#include <stdio.h>
#include <math.h>
double powerSeries(double x, int n);
int main(){
double x,sum;
int n,i;
printf("Enter x and n :");
scanf("%lf%d",&x,&n);
if (x<=-1 || x>=1)
  printf("x is out of range\n") ;
else if (n<0)
  printf("n is out of range\n") ;
else{
    sum=powerSeries(x,n);
      printf("power series for x=%.2f and n=%d is %.2f\n",x,n,sum);

}
return 0;
}

double powerSeries(double x, int n){
int i;
double sum;
sum=0;
for (i=0; i<=n; i++){
  sum = sum + pow(x,i) ;
}
return sum;
}
```

Question 6 : 17 points

Write a C program that models a guess-number game. The program stores a certain integer number as a declared constant, and the user (player) tries to guess this number. Assume that the number to be guessed is defined as a constant in your program. The program will guide the user throughout the game, such that, when the user enters a wrong number, the program will tell him or her whether the entered number (the user trial) is lower or higher than the number to be guessed. Also, the program allows the user to keep trying until he or she guesses the number correctly. After the user guesses the number correctly, the program prints the number of trials the user made to reach the correct number. Finally, the program asks the user whether he or she wants to replay the game by entering 'Y' or 'N'. Make sure that your program prints the appropriate required messages during the game.

Sample Run:

The following run of the program assumes that the number to be guessed is **100**.

```
Enter your guess:
80
Wrong Guess!
Hint: You entered a smaller
number!
Enter your guess:
120
Wrong Guess!
Hint: You entered a bigger number!
Enter your guess:
110
Wrong Guess!
Hint: You entered a bigger number!
Enter your guess:
100
You won!
You solved it in 4 trials!

Do you want to replay (Y/N)?
Y
Enter your guess:
70
Wrong Guess!
Hint: You entered a smaller
number!
Enter your guess:
100
You won!
You solved it in 2 trials!

Do you want to replay (Y/N)?
N
```

Key Solution

```c
/*This program models a guess number game*/
#include <stdio.h>
#define NUMBER_TO_BE_GUESSED 100
int main(){
    int userTrial;
    int trialCount;
    char replay;
do {

 trialCount = 0;
 do{

    printf("Enter your guess:\n");
    scanf("%d", &userTrial);
    trialCount++;

    if(userTrial != NUMBER_TO_BE_GUESSED){
        printf("Wrong Guess!\n");
        if(userTrial > NUMBER_TO_BE_GUESSED)
           printf("Hint: You entered a bigger number!\n");
        else
           printf("Hint: You entered a smaller number!\n");
    }
 } while(userTrial != NUMBER_TO_BE_GUESSED);
 printf("\nYou won! \n");
 printf("You solved it in %d trials!\n", trialCount)
 printf("\nDo you want to replay (Y/N)? \n");
 scanf("\n%c", &replay); // 1 point
}while(replay != 'n' && replay != 'N');
return (0);
}
```