KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

Information and Computer Science Department

2018 Summer Semester (Term 173)

ICS103 Computer Programming in C (2-3-3)

**Midterm Exam**

120 Minutes

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Student Name |  | | | | | | | | |
| KFUPM ID |  |  |  |  |  |  |  |  |  | |
| Class Section | DR. Mahmood Niazi SEC. 01 | | | □ UMTW 08:10 am | | | | |
| DR. Mahmnood Niazi SEC. 02 | | | □ UMTW 9.20 am | | | | |
| DR. Mahmood Niazi  SEC. 03 | | | □ UMTW 10:30 am | | | | |
| DR. FARAG AZZEDIN  SEC. 04 | | | □ UMTW 8:10 am | | | | |
| DR. FARAG AZZEDIN  SEC. 05 | | | □ UMTW 9:20 am | | | | |
| DR. FARAG AZZEDIN  SEC.. 06 | | | □ UMTW 10:30 am | | | | |

**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** | **20** |  |  |
| **2** | **15** |  |  |
| **3** | **18** |  |  |
| **4** | **20** |  |  |
| **5** | **10** |  |  |
| **6** | **17** |  |  |
| **Total** | **100** |  |  |

**Question 1: 20 points (2 marks each)**

1) Which one of the following C variable names is **not** valid?

1. \_kfupm3
2. \*kfupm
3. K2fpm\_26
4. All of the above are correct

2) Given the following declarations:

double a = 4.3265 ; int b = 72;

Which print statement produces the following output? (A square represents one space)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 4 | . | 3 | 3 |  | 7 | 2 |

1. printf("%7.2f%3d", a, b);
2. printf("%4.2f%4d", a, b);
3. printf("%5.2f%3d", a, b);
4. printf("%5.2f%2d", a, b);

3) Which of the following is a correct function prototype for a void function with two integer arguments?

A. void myfunction(int a, b);

B. myfunction (int a, int b);

C. void myfunction( int a, int b);

D. myfunction void( int a, int b);

4) Which of the following is **not** a reserved word in c:

|  |  |  |
| --- | --- | --- |
| 1. double | 1. if |  |
| 1. sqrt | 1. printf |  |

1. iii, iv
2. iii
3. iv, i
4. i, ii

5) Consider the following logical expression:

A || C + D && !E

Which of the following is equivalent to the above logical expression?

1. ((A || (C + D)) && !E)
2. (A || C) + (D && !E)
3. (A || ((C + D) && !E))
4. ((A || (C + D)) || E)

6) Which of the following is the correct order of evaluation for the C operators in the expression shown below?

a / b \* (c \* -d) % e

1. -, \*, /, \*, %
2. /, \*, %, -, \*
3. -, \*,%, /, \*
4. None of the above

7) What will be shown on the screen as a result of executing the following statements?

int a = 15, b = 10;

if (a == b)

b += 5;

a = a + 5;

printf("%d %d",a,b);

1. 15 10
2. 10 15
3. 20 10
4. 10 20

8) Which of the following is the correct 'C' code for the formula: root1 = 

1. root1 = (-b + sqrt(b\*b - 4\*a\*c)) / 2\*a
2. root1 = (-b + sqrt(b\*b - 4\*a\*c) / (2\*a))
3. root1 = (-b + sqrt(b\*b - 4\*a\*c)) / (2\*a)
4. root1 = -b + sqrt(b\*b - 4\*a\*c)/(2\*a)

9) To test if a given variable ***ch*** of type *char* contains a letter (lower case or capital), we use the condition:

1. ch >= 'a' && ch <= 'z' && ch >= 'A' && ch <= 'Z'
2. 'a' <= ch <= 'z' && 'A' <= ch <= 'Z'
3. ch >= 'a' && ch <= 'z' || ch >= 'A' && ch <= 'Z'
4. 'a' <= ch <= 'z' || 'A' <= ch <= 'Z'
5. 10) Consider the following C program fragment:

printf("%d", SPEED);

Which statement defines the **constant** SPEED with a value of 120?

1. int SPEED = 120;
2. #define SPEED = 120;
3. #define SPEED 120
4. SPEED = 120

**Question 2: 15 points**

Find the values of the following expressions**.** (13 points)

|  |  |
| --- | --- |
| expression | Value |
| 3>=5-1 | 0 |
| 8-4%3 | 7 |
| !6<=5-5 | 1 |
| 6+6/4.0 | 7.5 |
| 6!=2!=3 | 1 |
| 8-4.5/3 | 6.5 |
| 1!=2>=0 | 0 |
| 1==3<1 | 0 |
| (double)(13/2) | 6.0 |
| 1&&3==1 | 0 |
| 0||4>2 | 1 |
| 20 + 40 / 5 \* 3 | 44 |
| 2 < 4 || a == b && !c \*3 < 5 | 1 |

Use DeMorgan's theorem to find the equivalent of each logic expression

The **!** operator must NOT appear in the equivalent expression ( 2 points)

|  |  |
| --- | --- |
| logic expression | Equivalent expression |
| ! ( C != 3 ) | C==3 |
| ! ( B < 10 || B > 15 ) | B>=10 && B <= 15 |

**Question 3: 18 points**

What is the output of the following code fragments?

|  |  |
| --- | --- |
| Code fragment | Output |
| // 2 pts  int x = 3;  if(x/2\*2 == x)  printf("A");  else if(x/2\*3 == 2/2\*x)  printf("B");  else if(2/2\*x == x)  printf("C");  else  printf("D"); | B |
| // 2 pts  double grd = 60;  if(grd > 80)  printf("Excellent");  if(grd > 50)  printf("Good");  else  printf("Fail");  printf("\*\*\*"); | Good\*\*\* |

|  |  |
| --- | --- |
| // 2 pts  double x = 3.5;  switch (x){  case 2.5: printf("2");  break;  case 3.5: printf("3");  x=20;  case 4.5: printf("4");  break;  x=30;  default: printf("5"); x=40;}  printf("x=%f5.2", x); | Syntax error  Switch does not apply to double |
| // 4 pts  int fail = 0;  if (!fail || fail)  if (fail )  printf("D");  else  printf("C");  else  printf("A");  printf("B"); | CB |
| // 4 pts  int A = 0, B = 1;  if (A)  printf("A\n");  if (B)  printf("B\n");  if ( A = B)  printf("A=B\n");  else  printf("A!=B"); | B  A=B |
| // 4 pts  int i,j, cout = 0, cin=0;  for(j=8; j>5; j--) {  cout++;  for(i=1; i<=10; i++)  cin++;  }  printf("%d %d", cout, cin); | 3 30 |

**Question 4: [20 points]**

Part 1: (6 points)

Implement the following flowchart using nested if-statements. Do NOT make any changes to the flowchart. Do NOT write a full program. Just write the nested if-statements. (6 points)

**K< 89**

**K < 60**

**false**

**“middle”**

**“Low”**

**true**

**K>99**

**“Second”**

**“First”**

**false**

**K>70**

**“Third”**

**true**

**true**

**true**

**false**

**false**

if(k < 89)

    if(k < 60)

        printf("Low");

else

        printf("Middle");

else if(k > 99)

        printf("First);

else if(k > 70)

        printf("Second);

else

       printf("Third");

Part 2: (7 points)

Starting from variable declarations, write the necessary statements to count how may numbers from low to high (inclusive) are divisible by 5 but not 3. Assume that low and high are integer variables with given values and low < high.

For example if low=5 and high= 30, the numbers between 5 and 30 divisible by 5 but not 3 are:

5, 10, 20 and 25. 15 and 30 are not counted because they are divisible by 3. So the count is 4. No need to print the count.

int i,count;

count=0;

for(i=low;i<=high;i++)

if(i%5==0 && i%3!=0)

count++;

Part 3: (7 points)

Convert the following nested for loop into nested while loop

for(i=5;i>=2;i=i-1) {

k=0;

for(j=0;j<=10;j=j+1)

k=k+j;

printf("j=%d\n",j);

}

i=5;

while ( i >= 2)

{ k=0;

j=0;

while (j <= 10)

{ k=k+j;

j=j+1;

}

printf("j=%d\n",j);

i=i-1;

}

**Question 5: [10 points]**

An airline charges **zero** Saudi Riyals for the **first** 10 kilograms of baggage. For each extra kilogram of the **next** 25 kilograms the charge is 50.0 Saudi Riyals per kilogram. For extra kilograms above 35 kilograms, the charge is 55.0 Saudi Riyals per kilogram.

Write a function **baggeCharge** that receives the baggage weight in kilograms and returns the baggage charge in Saudi Riyals. Your function must be general and it must not contain **printf** and **scanf** statements.

Note: Don’t write the main function; write the function definition of **baggeCharge** only**.**

double baggeCharge(double weight)

{

double charge;

if(weight <= 10)

charge = 0;

else if(weight <= 35)

charge = (weight - 10) \* 50.0;

else

charge = 25 \* 50.0 + (weight - 35) \* 55.0;

return charge;

}

**Question 6: [17 points]**

An employee of a company is paid according to the formula shown below.

Write a complete C program that computes salary of N employees.

Your program will prompt for and read the number of employees N. if N, it will display an error message and terminates. If N is acceptable, your program will ask the user to enter the number of hours and the wage of each employee, then computes and displays his salary. If , for a certain employee, the number of hours or wage is zero or negative, the program displays an error message and moves to next employee without computing the salary for invalid hours or wage. Your program must work as shown in the sample runs below.

|  |
| --- |
| Enter number of employees: 5  Enter hours and wage for employee 1: 12 15.8  Salary = 189.60 SR for employee 1  Enter hours and wage for employee 2: -15 20.4  Error: Invalid input for employee 2  Enter hours and wage for employee 3: 34 5.8  Salary = 197.20 SR for employee 3  Enter hours and wage for employee 4: 19 -20.5  Error: Invalid input for employee 4  Enter hours and wage for employee 5: 23 15.8  Salary = 363.40 SR for employee 5 |

Sample run1: sample run2

|  |
| --- |
| Enter number of employees: -5  Invalid number of employees |

**#include <stdio.h>**

**int main(void) {**

**int hours,N,count;**

**double wage, salary;**

**printf("Enter number of employees: ");**

**scanf("%d",&N);**

**if(N<=0){**

**printf("Invalid number of employees\n");**

**return 0;**

**}**

**count = 1;**

**while (count<=N) {**

**printf("Enter hours and wage for employee %d: ",count);**

**scanf("%d%lf",&hours,&wage);**

**if(hours <= 0 || wage <= 0)**

**printf("Error: Invalid input for employee %d\n",count);**

**else {**

**if(hours <= 35)**

**salary = hours \* wage;**

**else**

**salary = 2 \* wage \* (hours - 35) + 35 \* hours;**

**printf("Salary = %.2f SR for employee %d\n", salary,count);**

**}**

**count = count+1;**

**}**

**return 0 ;**

**}**