12/12/2020

CSE1142 Midterm Exam – Fall 2020 (Online)
(Duration: 100 minutes)

PLEASE WRITE YOUR NAME, SURNAME, AND THE HONOR CODE BELOW IN YOUR HANDWRTING AND SIGN YOUR ANSWER SHEET.

HONOR CODE: "On my honor, I have neither given nor received any unauthorized and/or inappropriate assistance for this exam. The work done on this exam is totally my own. I understand that by the school code, violation of these principles will lead to a zero grade and is subject to harsh discipline issues."< your signature>

Submission Instructions:

- **You will write your answers on A4 size white blank sheets, with a legible handwriting.**
- **You will scan all your answer sheets using a (free) mobile phone scanner application into a single pdf file. You must use the PDF format.**
- **The name of the file should be "student name_surname.pdf".**
- **You have to upload your single pdf file to UES before the deadline.**
- **In case of last minute problems (in case of not being able to upload to UES), the answer sheets should be sent by email without exceeding the deadline. (If you cannot upload it to UES, you can upload it to Canvas without exceeding the deadline).**
- **The answer sheets sent after the deadline will not be accepted.**
- **The maximum number of uploads is limited to 2.**

Q.I (20 points)

a) (10 pts) Suppose that the following program prints

```
Exception Message 1
Exception Message 2
Exception Message 3
```

on console screen. Please fill in the blanks in the **System.out.println** statements (*from lines A to F*) below to get this output. If you think that a **System.out.println** is not executed, then you can fill as **"Not executed!"**.
Please write Q1.lineX (X can be from A to F) and your corresponding answers to the sheet.

```
public class Test {
     public static void main(String[] args) throws Exception {
          try {
               myMethod(7);
          } catch (NumberFormatException e) {
/*line A*/      System.out.println("____________________");
          }
     }
     public static void myMethod(int n) throws Exception {
          try {
               if (n > 0)
                    throw new NumberFormatException();
               else if (n < 0)
                    throw new Exception();
               else
/*line B*/           System.out.println("____________________");
/*line C*/           System.out.println("____________________");
          } catch (Exception e) {
/*line D*/           System.out.println("____________________");
          } finally {
/*line E*/           System.out.println("____________________");
          }
/*line F*/ System.out.println("____________________");
     }
}
```

b) (10 pts) What does the class Test below print out if it is compiled and run?

```
class A {
    A(){
        System.out.println("A");
    }
    A(String name){
        this();
        System.out.println("A: " + name);
    }
   public void print(A a) {
        System.out.println("printA 1");
    }
}
class B extends A {
    B(){
        System.out.println("B ");
    }
    B(String name){
        this();
        System.out.println("B: " + name);
    }
    public void print(A a) {
        System.out.println("printB 1");
    }
    public void print(B b) {
        System.out.println("printB 2");
    }
}

public class Test {
    public static void main (String [] args) {
        A var1 = new B("CSE");
        B var2 = new B();
        A var3 = new A();


        ((B)var1).print(var3);
        var2.print(var2);
        var2.print(var3);
        var3.print(var3);
    }
}
```

Q.II(18 points) Design a class named **Square**, a subclass of **GeometricObject**. Please fill in the constructor and methods as well as the class definition below.

```
public class ____________________________________ {
  double side;
  public Square(double side) {



  }
  public _______ getArea() {









  }
  public _______ getPerimeter() {









  }
  /* Override the equals method.
     Square s1 is said to be equal to Square s2 if s1.side == s2.side.
  */
  public boolean equals(Object o) {









  }
}
```

Q.III (46 points)

a) (4 points) Analyze the following code segments:

Code 1:

```
int even;
if (number % 2 == 0)
    even = 1;
else
    even = 0;
```

Code 2:

```
int even = (number % 2 == 0);
```

Will these two code segments *always* produce the same assignments for **even** in the end? If you believe that they will behave the same way, provide a convincing argument in 1-2 sentences to explain why you believe that this is so. If you believe that they will behave differently, provide an example to explain where they behave differently and describe the difference.

b) (8 points) Write a macro named **DIVIDE** below to divide given two numbers. Then, show the output of the following code:

```
#define DIVIDE_____________________________

enum hello{x,y=19,z,t=-5};

int main(){

     printf("%d\n%d\n%d\n%d\n",x, y, z, t);
     printf("%d", DIVIDE(3+3, 1+1));
     return 0;
}
```

c) (14 points) Show the output of the following code:

```
int value = 10;
int p1(int *input){
      *input = *input * value;
      printf("In p1: %d\n", value);
      return value;
}
int p2(int input){
      int value = 5;
      value = input / value;
      printf("In p2: %d\n", value);
      return value;
}
int p3(int input){
      static int value = 5;
      value = ++input + value;
      printf("In p3: %d\n", value);
      return value;
}
int main()  {
      printf("Main before p1: %d\n", value);
      value = p1(&value);
      printf("Main before p2: %d\n", value);
      value = p2(value);
      printf("Main before p3: %d\n", value);
      value = p3(value);
      printf("Main at the end: %d\n", value);
      printf("-----\n");
      int value = 30;
      printf("Main before p1: %d\n", value);
      value = p1(&value);
      printf("Main before p2: %d\n", value);
      value = p2(value);
      printf("Main before p3: %d\n", value);
      value = p3(value);
      printf("Main at the end: %d\n", value);

}
```

d) (20 points) There are a total of 10 errors in the following C program. Please indicate the line number and your correction in the corresponding statement.

```
/*ln.1*/  #include (stdio.h)
/*ln.2*/  #Define PI 3.14
/*ln.3*/  int main {
/*ln.4*/    Int radius, width, height;
/*ln.5*/    Float area, perimeter;
/*ln.6*/    printf("\nEnter radius of a circle: ");
/*ln.7*/    scanf("%d", radius);
/*ln.8*/    area = PI * radius * radius;
/*ln.9*/    printf("\nArea of circle : %d ", area);
/*ln.10*/   perimeter = 2 * PI * radius;
/*ln.11*/   printf("\nPerimeter: %f ", perimeter)
/*ln.12*/   printf("\nEnter the width of a rectangle : ");
/*ln.13*/   scanf("%d", &width);
/*ln.14*/   printf("\nEnter the height of a rectangle : ");
/*ln.15*/   scanf("%d", &height);
/*ln.16*/   area = width * height;
/*ln.17*/   printf("\nArea of rectangle : %f", area);
/*ln.18*/   Return 0;
```

Q.IV (16 points) In this problem, you will write a program that prompts the user to enter an integer number (N) and then displays the following N-by-N square pattern. For a given size N=5, the pattern should be as follows:

```
* * * * *
+ * * * *
+ + * * *
+ + + * *
+ + + + *
```

Please complete the given implementation below:

```
int main(){
     int N;

     printf("Enter an integer for N: ");
     scanf(_________________);
    for (_____________________________) {

       for (_____________________________){
            // print + in the loop

       }

       for (_____________________________){
            // print * in the loop

       }

       // print a new line
       printf("\n");
    }
}
```