Final Exam

Started: May 17 at 9:35pm

Quiz Instructions

- This exam covers all the modules with the concentration on queue, recursive and exception handling.
- There will be no make up exam.
- Check the spelling and the syntax of the code
- programming questions will be in the form of the fill-in-blank
- you will not see the questions that you answered wrong after you submit the exam.
- you will be able to see the questions you missed after everyone has taken the test
- feel free to use the JGrasp to code
- if you think the answer to the fill in the blank questions for example is : a - 1 , make sure to enter a, -, 1 in the separated provided boxes.
- if the answer for example is -1: enter the -1 in the provided box

---

Question 1 — 1 pts

The answers to all the questions is my own work and I did not get help from any other resources while taking the exam. Using resources while taking the exam is considered plagiarism and will lead to a grade zero. Any academic dishonesty will be reported.

- (●) I agree
- ( ) I disagree

Question 2 — 3 pts

The following code is causing an exception during the runtime.

```
class Mystery
{
 public static void main(String[] args)
 {
    Object  b1 = new Scanner(System.in);
    Object b2 = new Random();
    doSomething (b1);
    doSomething(b2);

 }
 public static void doSomething(Object o)
 {

   Scanner  s = (Scanner)o;
   System.out.print("Enter your name: ");
   String name = s.nextLine();
   System.out.println(name.length());
 }
}
```

which one of the following is the correct answer to fix the problem

A. Exception can be prevented by adding an if statement and type casting in the method doSomething
B. Exception can be handled by adding try-catch in the main method
C. answers A and B both are correct
D. Non of the answers

- ○ B
- ○ D
- ○ A
- ◉ C

Question 3 — 2 pts

A recursive method with no base case

- ○ all of the answers
- ○ generates many lines of output
- ○ does not generate any output
- ◉ causes an infinite recursion

Question 4 — 4 pts

The method back gets an array of chars and an index as its parameter and displays the content of the array backward recursively with every letter printed 2 times. for example if the array letters contains:

```
letters = {'2','2','0','2',' ', 'g', 'n', 'i', 'r', 'p', 's'  };
then the output with the call back(letters, letters.length -1) should be: sspp
rriinngg  22002222
```

```
 Fill in the blank to complete the recursive code
NO FOR LOOP can be used
```

```
public static void back(char[] a, int pos)
{
    if( pos <= -1 )
        return ;

    System.out.print(a[ pos ]+""+a[ pos ] );

    back( a , pos - 1 );
}
```

if the answer for one of the box

Question 5

3 pts

The following method is a recursive method. This method gets two integer numbers as its parameters(called num and factor, then the method finds the biggest factor of the parameter num. The biggest factor of a number cannot be the number itself. for example the biggest factor of 12 is not 12 and it is 6.

call mystery1(186,185) will display : The greatest factor is: 93

the call mystery1(133, 132) will display: The greatest factor is: 19

The way the method works is : 133 % 132 is not zero, 133% 131 is not zero, 133% 130 is not zero, the process will continue until we get to 133 % 19 is zero then the recursive call ends.

Note: if you need to enter for example a + 1, make sure to enter a, + , and 1 in the separate provided boxes.

```
public static void mystery1(int num, int factor)
 {
    if (factor > 0 && num % [ factor ] == [ 0 ] )
    {
      System.out.println("The greatest factor is: " + [ factor ] );
      [ return ] ;
   }
```

mystery1 ([num] , [factor] [-]

1);

}
}

Question 6 — 4 pts

The following recursive method has been created. This method accepts two integer parameters called a and b respectively.

Assume that the **name of the method is m** and the code in the recursive method is:

```
if ( a > b )
   return;
if(a % 4 == 3)
   System.out.print(a+ " ");
 m(a + 3, b);
```

what would be the output with the following call m(4,50) where a is 4 and b is 50

Enter your answers separated with only one space

7 | 19 | 31 | 43

Question 7 — 3 pts

The following recursive method **called z** is created. This method accepts two parameters: **A string s, and an integer index**

```
The code in the method is:

    if (index == s.length())
```

```
        return "";          <------ base case

     if(index % 2 != 0)
        return ""+ s.charAt(index) + z(s,index+1);          <---- recursive c
all

     else
        return z(s,index+1);             <----- recursive call
```

```
What would be the output with the call:  System.out.println( z("javajavajava",
0));
```

Enter your answer. aaaaaa

Question 8

2 pts

Assume that you have written a try-catch block in your code. What will happen if an error occurs in the try block?

- ○ the catch blcck will never be executed
- ○ The next line of the code will be executed in the try block
- ○ the code in the try block will be executed, then the code in the catch block will be excuted
- ◉ Flow control jumps to the catch block

Question 9

2 pts

```
The following code segment has been created . What would be the output if we w
ere to create a program and run it
     int[] a = {1,2,3,4,5};
     try
      {
         for(int i = 0; i <= a.length; i++)
            a[i]++;

         int ac = 10;

         int b = ac/0;
```

```
    }
  catch(ArrayIndexOutOfBoundsException e)
    {
      System.out.print( "Index out of bound ");
    }
   catch(Exception e)
   {
     System.out.print("division by zero  ");
   }
```

- index out of bound division by zero
- none of the answers
- Index out of bound
- division by zero

Question 10

2 pts

The following code segment has been created. the code can be placed in a method.

The code keeps asking the user to enter a double value until the user provides a valid input. Once a valid input is entered the loop ends and the method returns the value

However it has an infinite loop. Fill in the blank so that the method works.

```
      int done = 0;
      double input = 0;
      while (done!= -1)
      {
         try
         {
            System.out.print("Enter an double: ");
            input = kb.nextDouble();
```

done

=

-1

;

```
         }
         catch(Exception e)
         {
            System.out.println("You did not enter a valid input ");
```

```
        }
        kb.nextLine();
      }
      return input;
```

Question 11

2 pts

The following code is not compiling. . Fill in the blank so that the code does not have any syntax errors

```
public static void e(int a) throws

Exception


{
      if( a < 0)
        throw new Exception();
}

Check the spelling and the syntax of your answers
```

Question 12

2 pts

A queue is a _____ data structue

- there is no order of which element can go out first
- (selected) FIFO
- none of the answers

- FILO

Question 13

2 pts

The remove method in the queue class, removes the element from the

- front of the list
- any where in the list depending on the index
- end of the list
- middle of the list

Question 14

2 pts

which one of the following is true

A. Java Queue class is an interface
B. Java LinkedList class has implemented the Queue interface
C. An object of type Queue can be declared
D. An object of type Queue cannot be instantiated as a Queue object
E. An Object of type Queue can be instantiated as a LinkedList object
F. All the answers are correct

- ○ B
- ○ A
- ○ D
- ◉ F
- ○ E
- ○ C

Question 15

2 pts

Which one of the following are the operations that can be done with a queue of objects

A. dequeue
B. enqueue
C. push
D. pop
E. answers A and B

- ○ C
- ○ A
- ○ B

- (●) E
- ( ) D

Question 16

4 pts

Consider the following code segment , The variable **q is an object of type Queue**, the variable **s is an object of type Stack**.

- peek method looks at the first element in the queue without removing it.
- the remove method removes the first element from the queue.
- add method adds an element to the and of the queue or add an element to the top of the stack.
- pop method removes an element from the top of the stack

What would be the content of the variable **q** after we complete the second while loop in the code

```
for (int i = 40; i <= 65; i+=3)
    {
      if(i % 5 == 0)
        q.add(i);
  }
    while (!q.isEmpty()) {
      s.add(q.peek());
      s.add(q.peek());
      q.remove();
      System.out.print("*");
    }
    while (!s.isEmpty()) {
     q.add(s.pop());
    }
   //what is the content of the q if we were to display it here
```

Answer: 55 | 55 | 40

40

Quiz saved at 10:02pm

Submit Quiz