Quiz 1

Covers: Weeks 1 - 9

1. What is the output of the following program?

```
public class ZeroDiv{

public static void main(String args[]){
    int x = 0, y = 10;
    try {
        y /= x; }
        System.out.print("/ by 0"); catch(Exception e) {
        System.out.print("error");
        }
    }

}
```

a) 0
b) Error
c) Compilation fails
d) An uncaught exception is thrown at runtime
e) No output

1. What is the output of the following program?

Can’t have a statement between try and catch

```
public class ZeroDiv{

public static void main(String args[]){
    int x = 0, y = 10;
    try {
        y /= x; }
        System.out.print("/ by 0"); catch(Exception e) {
        System.out.print("error");
        }
    }

}
```

a) 0
b) Error
c) Compilation fails
d) An uncaught exception is thrown at runtime
e) No output

2. Which of the following statements about exception handling in Java are true?

a) “throw” can be used to declare an exception in a method, and the exception will be thrown in this method

b) “throws” is used to throw the exception objects

c) “try” is used to detect if there is an exception in its block and if so it intercepts the exception and execute the code in the “catch” block

d) No matter if there is an exception or not, the code in the “finally” block will be executed

e) You cannot throw an exception in a “try” block

2. Which of the following statements about exception handling in Java are true?

a) “throw” can be used to declare an exception in a method, and the exception will be thrown in this method

b) “throws” is used to throw the exception objects

c) “try” is used to detect if there is an exception in its block and if so it intercepts the exception and execute the code in the “catch” block

d) No matter if there is an exception or not, the code in the “finally” block will be executed

e) You cannot throw an exception in a “try” block

3. What is the output of the following code?

a) Compilation error
b) eE
c) Ee
d) eE1eE3eE5
e) Ee1Ee3Ee5

```
public class Test {

    private static void test(int[] arr) {
        for (int i = 0; i < arr.length; i++) {
            try {
                if (arr[i] % 2 == 0) {
                    throw new NullPointerException();
                } else {
                    System.out.print(i); }
                } finally {
                    System.out.print("e"); }
                }
            }

    public static void main(String[] args) {
        try {
            test(new int[] {0, 1, 2, 3, 4, 5});
        } catch (Exception e) {
            System.out.print("E");
        }
    }
}
```

3. What is the output of the following code?

a) Compilation error

b) eE

c) Ee

d) eE1eE3eE5

e) Ee1Ee3Ee5

```
public class Test {

    private static void test(int[] arr) {
        for (int i = 0; i < arr.length; i++) {
            try {
                if (arr[i] % 2 == 0) {
                    throw new NullPointerException();
                } else {
                    System.out.print(i); }
                } finally {
                    System.out.print("e"); }
                }
            }

    public static void main(String[] args) {
        try {
            test(new int[] {0, 1, 2, 3, 4, 5});
        } catch (Exception e) {
            System.out.print("E");
        }
    }
}
```

An exception is a type of object which can be thrown by any other class/method. Here method test throws the null pointer exception, the finally block is always executed in a try-catch block so “e” is printed, and then in main the null pointer exception is caught resulting in printing “E”

4. What is the output of the following code?

```
public class Test {

    public static void main(String[] args) {
        System.out.println("return value of getValue(): "
                           + getValue());
    }

    public static int getValue() {
        try {
            return 0;
        } finally {
            return 1;
        }
    }

}
```

4. What is the output of the following code?

```
public class Test {

    public static void main(String[] args) {
        System.out.println("return value of getValue(): "
                           + getValue());
    }

    public static int getValue() {
        try {
            return 0;
        } finally {
            return 1;
        }
    }

}
```

return value of getValue(): 1

5. Which is the correct statement about exception handling in Java?

a) If you have defined a possible exception in a method with “throw” you definitely have this exception when you use the method
b) If there is no exception thrown in the “try” block then the code in the “finally” block won’t be executed
c) If your program throws an exception then there must be an error in your program – you need to debug and fix the error
d) An unchecked exception in Java derives from RuntimeException or its children

5. Which is the correct statement about exception handling in Java?

a) If you have defined a possible exception in a method with “throw” you definitely have this exception when you use the method
b) If there is no exception thrown in the “try” block then the code in the “finally” block won’t be executed
c) If your program throws an exception then there must be an error in your program – you need to debug and fix the error
d) An unchecked exception in Java derives from RuntimeException or its children

6. You don’t need a “finally” block to release resources if you wrote your “try” block as follows:

```
try (// Acquire resources here) {
    ...
} catch ... {
}
```

a) True

b) False

6. You don’t need a “finally” block to release resources if you wrote your “try” block as follows:

```
try (// Acquire resources here) {
    ...
} catch ... {
}
```

a) True

b) False

7. Which of the following is incorrect:

a) A “try” block cannot be omitted
b) Multiple “catch” blocks can be used
c) A “finally” block can be omitted
d) A “finally” block can be used without any “try” or “catch” block

7. Which of the following is incorrect:

a) A “try” block cannot be omitted
b) Multiple “catch” blocks can be used
c) A “finally” block can be omitted
d) A “finally” block can be used without any “try” or “catch” block

8. Suppose your program reads a value entered by the user. How should you create a custom exception that is thrown if the input is greater than 10?

a) if (i > 10) throw new Exception("something's wrong!");
b) if (i > 10) throw Exceptione("something's wrong!");
c) if (i > 10) throw new Exceptione("something's wrong!");
d) if (i > 10) throw Exception("something's wrong!");

8. Suppose your program reads a value entered by the user. How should you create a custom exception that is thrown if the input is greater than 10?

a) if (i > 10) throw new Exception("something's wrong!");
b) if (i > 10) throw Exceptione("something's wrong!");
c) if (i > 10) throw new Exceptione("something's wrong!");
d) if (i > 10) throw Exception("something's wrong!");

9. In the program below where will the references to the variables a, b, and c, be stored?

a) Heap, heap, heap

b) Heap, stack, heap

c) Heap, stack, stack

d) Heap, heap, stack

```
class A {
    private String a = "aa";

    public boolean methodB() {
        String b = "bb";
        final String c = "cc";
    }
}
```

9. In the program below where will the references to the variables a, b, and c, be stored?

a) Heap, heap, heap

b) Heap, stack, heap

c) Heap, stack, stack

d) Heap, heap, stack

```
class A {
    private String a = "aa";

    public boolean methodB() {
        String b = "bb";
        final String c = "cc";
    }
}
```

"aa", "bb" and "cc" are all in the heap but we are talking about references to these values.

10. What characterizes the Set interface?

a) A set is a collection of elements which contains elements along with their key
b) A Set is a collection of elements which contains hashcode of elements
c) A set is a collection of elements which cannot contain duplicates
d) A set is a collection that is always ordered

10. What characterizes the Set interface?

a) A set is a collection of elements which contains elements along with their key
b) A Set is a collection of elements which contains hashcode of elements
c) A set is a collection of elements which cannot contain duplicates
d) A set is a collection that is always ordered

11. What is the parent class of Error and Exception classes?

a) Throwable
b) Catchable
c) MainError
d) MainException

11. What is the parent class of Error and Exception classes?

a) Throwable
b) Catchable
c) MainError
d) MainException

12. Which arithmetic operations can (together or alone) result in the throwing of ArithmeticException?

a) /, %
b) *. +
c) !, -
d) <<, >>

12. Which arithmetic operations can (together or alone) result in the throwing of ArithmeticException?

a) /, %
b) *. +
c) !, -
d) <<, >>

13. The following are descriptions about List interface, Set interface and Map interface, which is false?

a) They all inherit from the Collection interface
b) List is an ordered interface, so we can control precisely where each element is inserted when using the interface
c) Set is a collection that does not contain duplicate elements
d) Map provides a mapping from key to value and Map cannot contain the same key several times, each key can only map a value

13. The following are descriptions about List interface, Set interface and Map interface, which is false?

a) They all inherit from the Collection interface
b) List is an ordered interface, so we can control precisely where each element is inserted when using the interface
c) Set is a collection that does not contain duplicate elements
d) Map provides a mapping from key to value and Map cannot contain the same key several times, each key can only map a value

14. Of the following statements about the Collections class, which is **false**?

a) Both ArrayList and LinkedList implement the List interface
b) Access to elements of an ArrayList is faster than elements of a LinkedList
c) When adding and removing elements, ArrayList performs better than LinkedList
d) HashMap implements the Map interface, which allows any type of key and value object and allows null to be used as a key or value

14. Of the following statements about the Collections class, which is **false**?

a) Both ArrayList and LinkedList implement the List interface

b) Access to elements of an ArrayList is faster than elements of a LinkedList

c) When adding and removing elements, ArrayList performs better than LinkedList

d) HashMap implements the Map interface, which allows any type of key and value object and allows null to be used as a key or value

Performs worse because potentially will need to shift elements to make a slot

15. Which of the following interfaces are directly inherited from the Collection interface (multiple answers)?

a) List
b) Map
c) Set
d) Iterator

15. Which of the following interfaces are directly inherited from the Collection interface (multiple answers)?

a) List
b) Map
c) Set
d) Iterator

16. The following is statement is about Collection and Collections, which is **true**?

a) Collection is a class under java.util which contains various static methods for collection operations
b) Collection is a class under java.util which is the parent interface of various collection structures
c) Collections is a class under java.util which is the parent interface for the various collection structures
d) Collections is a class under java.util which contains various static methods for collection operations

16. The following is statement is about Collection and Collections, which is **true**?

a) Collection is a class under java.util which contains various static methods for collection operations

b) Collection is a class under java.util which is the parent interface of various collection structures

c) Collections is a class under java.util which is the parent interface for the various collection structures

d) Collections is a class under java.util which contains various static methods for collection operations

17. What will be printed out on running the following program

a) 2,2
b) 2,3
c) 3,2
d) 3,3

```
public class Test{
    public static void main(String [] args){
        List list=new ArrayList();
        list.add("a");
        list.add("b");
        list.add("a");

        Set set=new HashSet();
        set.add("a");
        set.add("b");
        set.add("a");

        System.out.println(list.size()+","+set.size());
    }
}
```

17. What will be printed out on running the following program

a) 2,2

b) 2,3

c) 3,2

d) 3,3

```
public class Test{
    public static void main(String [] args){
        List list=new ArrayList();
        list.add("a");
        list.add("b");
        list.add("a");

        Set set=new HashSet();
        set.add("a");
        set.add("b");
        set.add("a");

        System.out.println(list.size()+","+set.size());
    }
}
```

18. After the code below is executed what are the elements in NumberList (if any)?

```
List<Integer> NumberList = new ArrayList<Integer>();
NumberList.add(2);
NumberList.add(4);
NumberList.add(1);
NumberList.add(3);
NumberList.add(5);
for(int i = 0; i < NumberList.size(); ++i) {
    int v = NumberList.get(i);
    if(v%2 == 0) {
        NumberList.remove(v);
    }
}
System.out.println(NumberList);
```

a) 2,4,1,3,5

b) 2,1,3,5

c) 4,1,3,5

d) There will be an out of bounds exception

18. After the code below is executed what are the elements in NumberList (if any)?

```
List<Integer> NumberList = new ArrayList<Integer>();
NumberList.add(2);
NumberList.add(4);
NumberList.add(1);
NumberList.add(3);
NumberList.add(5);
for(int i = 0; i < NumberList.size(); ++i) {
    int v = NumberList.get(i);
    if(v%2 == 0) {
        NumberList.remove(v);
    }
}
System.out.println(NumberList);
```

a) 2,4,1,3,5

b) 2,1,3,5

c) 4,1,3,5

d) There will be an out of bounds exception

19. A local variable in a method can be public.

a) True
b) False

19. A local variable in a method can be public.

a) True
b) False

20. What will happen when the following code is executed?

```
class MyError extends Error{ }

public class TestError {
    public static void main(String args[]) {
        try {
            test();
        } catch(Error ie) {
            System.out.println("Error caught");
        }
    }

    static void test() throws Error {
        throw new MyError();
    }
}
```

a) Runtime error test() method does not throw an error type instance
b) Compile time error Cannot catch Error type objects
c) Compile time error Error class cannot be extended
d) Prints “Error caught”

20. What will happen when the following code is executed?

```
class MyError extends Error{ }

public class TestError {
    public static void main(String args[]) {
        try {
            test();
        } catch(Error ie) {
            System.out.println("Error caught");
        }
    }

    static void test() throws Error {
        throw new MyError();
    }
}
```

a) Runtime error test() method does not throw an error type instance
b) Compile time error cannot catch Error type objects
c) Compile time error Error class cannot be extended
d) Prints “Error caught”

21. Annotations about annotations are called:

a) Depreciations
b) Metadata
c) Meta-annotations
d) There is no such thing

21. Annotations about annotations are called:

a) Depreciations
b) Metadata
c) Meta-annotations
d) There is no such thing

22. If a method is tagged with @Deprecated then:

a) Nothing special happens it is just information for javac
b) Javac will issue a warning but still generate the .class file
c) Javac ends in an error

22. If a method is tagged with @Deprecated then:

a) Nothing special happens it is just information for javac
b) Javac will issue a warning but still generate the .class file
c) Javac ends in an error

It will "throw a warning if someone uses this method" but nothing happens when compiling the class where the annotation is added.

23. If a Java interface defines two or more methods you cannot use lambda expressions with it.

- True
- False

23. If a Java interface defines two or more methods you cannot use lambda expressions with it.

- True
- False

24. In a Graphical User Interface, you can only store a container in a different type of container (for instance you can only add a vertical box to a grid or a horizontal box but not another vertical box).

a) True
b) False

24. In a Graphical User Interface, you can only store a container in a different type of container (for instance you can only add a vertical box to a grid or a horizontal box but not another vertical box).

a) True
b) False

25. If you are importing a .css file (style sheet) in a JavaFx application and the file cannot be found then:

a) If you haven't inserted the CSS file loading into a try .. catch .. block the application crashes
b) There is no exception thrown at the JavaFx application level. The application doesn't crash without a try .. catch .. block but it exits
c) There is no exception thrown at the JavaFx application level. The application writes a warning to the console and continues with default styling.

25. If you are importing a .css file (style sheet) in a JavaFx application and the file cannot be found then:

a) If you haven't inserted the CSS file loading into a try .. catch .. block the application crashes

b) There is no exception thrown at the JavaFx application level. The application doesn't crash without a try .. catch .. block but it exits

c) There is no exception thrown at the JavaFx application level. The application writes a warning to the console and continues with default styling.

26. A JavaFx application must have an init(), start() and stop() method.

a) True
b) False

26. A JavaFx application must have an init(), start() and stop() method.

a) True
b) False

Only start() is abstract in the Application class and has to be rewritten.