**CSC 142 Exam #3 Winter, 2019**

1. A ‘smiley face’ emoji is:
2. An ASCII character
3. A Unicode character
4. Both a.) and b.)
5. Neither a.) nor b.)

b.

1. (T/F): When a method is called, the value of each argument is passed to its matching parameter, and separate memory space is allocated to store the parameter value. This way of passing the value of arguments is called ***pass-by-reference***.

**False: *pass-by-value***

1. Multiple methods of a class can share the same name as long as **one** of the following two rules are met.

Rule 1: \_They have different number of parameters\_\_\_\_\_\_\_\_\_\_\_\_

Rule 2: \_\_parameters are of different type when number of parameters are same\_\_\_\_\_\_\_\_\_

1. Assume that the following array has been declared, though you do not necessarily know its length.

int [ ] numbers = new int [ ??] ;

What expression should be used to:

1. access the first element? int[0]
2. access the last element, assuming the array has 10 elements?

int[9]

1. access the last element, regardless of the size of the array? Int[-1]
2. Can we invoke one constructor from another constructor of the same class? If yes, how is it done?

Yes, by using the reserve word: ***this***

1. Two methods in the same class with the same name are called

\_\_\_\_overloaded\_\_\_\_\_\_\_ methods.

1. A method that does not have a method body is called a

\_\_\_\_\_abstract\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ method.

1. When the following lines are executed, how many objects are created?

String str1 = “KungFu”;

String str2 = “KungFu”;

String str3 = “KungFu”;

***ONE***

1. a.) Is the Math class an abstract class? **No**

b.) Can you create an instance of the Math class? **No**

1. Why does a method to swap two array elements work correctly when a method to swap two integer values does not?
2. **Unlike integers, arrays are objects and use reference semantics.**
3. Because of a bug in Java.
4. Because arrays are slower and more inefficient than primitives, so swapping is possible.
5. Changes to an array parameter's elements will not be seen in the original array by the caller.
6. Because arrays use more memory than primitive values.
7. *Given the following method:*

     public static void newArray(long [] number)

    {

         number = new long [25];

         number [0] = (long) 85.0;

    }

*What is the output after the following code is run?*

    long [] numberTwo = new long [25];

   numberTwo[0] = 13;

    newArray(numberTwo);

    System.out.println("numberTwo[0] = " + numberTwo[0]);

***13***

1. Which of the following constructors are invalid?
2. public int ClassA (int one){

// code

}

1. public ClassB (int one, int two){

// code

}

1. void ClassC () {

//code

}

**a**

1. What is the problem with this code?

**public** **class** Person {

**private** String fullName;

**public** Person(){

fullName = “ “;

**this**("unassigned");

}

**public** Person(String name){

fullName = name;

}

}

1. Suppose a method in the BankAccount class is defined as:

public double computeInterest(int rate)

And suppose the client code has created a BankAccount object named acct.

Which of the following would be a valid call to the above method?

1. Top of Form
2. Bottom of Form
3. acct.computeInterest(42.0, 15);
4. new BankAccount(42).computeInterest();
5. double result = acct.computeInterest(42);
6. double result = computeInterest(acct, 42);

***C***

1. What is the output of the following code fragment?

**public** **class** Account {

**private** **static** **double** *balance*= 0;

**public** **static** **double** getBalance(){

**return** *balance*;

}

**public** **double** add(**double** deposit)

{

**return** deposit+= *getBalance*();

}

**public** **static** **void** main(String[] args) {

Account account = **new** Account();

**double** currentBalance;

currentBalance = account.add(100.0);

System.***out***.println("balance is:" + *getBalance*());

System.***out***.println("Current balance is:" + currentBalance);

}

}

***0***

***100.0***

1. In Java, single text characters are represented by what data type?

**char**

1. Which of the following statements about constructors are true? Circle all that apply.
2. A class can have many methods but only one constructor.
3. A constructor can be used instead of fields to represent the data inside a class.
4. A constructor wastes memory in the computer so it should be used sparingly.
5. **A constructor is a special method that creates an object and initializes its state.**
6. Constructors are considered bad programming style and should be avoided.
7. **A constructor is the code that is called when you use the ‘new’ keyword.**
8. What is wrong with the following declaration?

class Table {

Person student = new Person();

public void getStudent() {

return student;

}

// other methods go here…

}

1. Rewrite the following constructors, so the first one calls the second one:

public ClassOne(int alpha) {

this.alpha = alpha;

this.beta = 0;

}

public ClassOne(int alpha, int beta) {

this.alpha = alpha;

this.beta = beta;

}

1. Why is the code below showing a compile time error?

class X {

private int m = 48;

}

class Y extends X {

void methodOfY() {

System.out.println(m);

}

}

1. What is printed by the following program?

public class Mystery {

public static void main(String[] args) {

String one = “two”;

String two = “three”;

String three = “1”;

int number = 20;

sentence(one, two, 3);

sentence(“eight”, three, number / 2);

}

public static void sentence(String three, String one, int number){

System.out.println(one + “ times “ + three + “ = “ + (number \* 2);

}

}

1. Suppose **Car** and **Motorcycle** are subclasses of **Vehicle**. Which of these declarations are ***allowed***?
2. Car jaguar = **new** Vehicle();
3. Vehicle ferrari = **new** Car();
4. Motorcycle ducati = **new** Vehicle();
5. Motorcycle goldwing = **new** Car();