1. *(2 points)* What is the decimal value of a binary literal 0b11010 in a Java program?

Decimal value: **26**

1. *(2 points)* List four primitive types and four classes used in many Java programs:

**Primitive: char, byte, short, int, long, boolean, float, double**

**Classes: System, String, Scanner, PrintWriter, Math, Arrays, etc.**

1. *(3 points)* Write the Java code tocreate a **Scanner** to read the **keyboard**. Don’t worry about imports, or using or closing the Scanner.

 **Scanner keyboard = new Scanner(System.in);**

1. *(2 points)* Write one Java statement to print four variables of type **char**, **String**, **int**, and **double**, named *myChar*, *myString*, *myInteger*, and *myDouble*, separated by commas. The double must be printed with 5 digits after the decimal point.

**System.out.printf(“%c,%s,%d,%.5f\n”,**

**myChar, myString, myInteger, myDouble);**

1. *(2 points)* When **declaring** a Java method, is the programmer required to explicitly specify the **data** **type** of each parameter?
	1. **yes**
	2. no
2. *(1 point)* What is the value of the **boolean** variable **myBoolean** after the following statement?

 **boolean** myBoolean = (16 <= 5) && true; **false**

1. *(3 points each)*Write the output of the following code in the spaces provided.

HINT: Consider integer math, type casting, order of operations. Indices are zero based!

**public** **class** FinalExam1 {

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

 **int** i = 23, j = 4;

 **double** x = 1.3, y = 5.2;

 // First line

 System.***out***.printf("%d,%.2f\n", i / j + i % j, j \* x);

 // Second line

 System.***out***.printf("%d,%.2f\n", (**int**)x - j, y + x \* 2);

 String s0 = "Java";

 String s1 = "~!@#$%^0123456789";

 // Third line

 String str = s1.charAt(5) + ":" + (s0 + s1).charAt(7);

 System.***out***.println(str);

 // Fourth line

 **int** num = s1.indexOf('2') + s1.indexOf('@');

 System.***out***.printf("%d\n", num);

 // Fifth line

 String sub = s0.substring(1, 3) + s1.substring(2, 5);

 System.***out***.println(sub);

 }

}

**First line of output: 8,5.20**

**Second line of output: -3,7.80**

**Third line of output: %:#**

**Fourth line of output: 11**

**Fifth line of output: av@#$**

1. *(3 points each)*Write the output of the following code in the spaces provided.

HINT: Consider pass by value, pass by reference, immutability of strings.

**import** java.util.Arrays;

**public** **class** FinalExam2 {

 **public** **static** **void** capitalize(String sChars) {

 sChars = sChars.toUpperCase();

 System.***out***.println(sChars); // First line

 }

 **public** **static** **void** square(**double** dValue) {

 dValue = Math.*pow*(dValue, 2);

 System.***out***.println(dValue); // Second line

 }

 **public** **static** **void** sort(**int**[] iArray) {

 Arrays.*sort*(iArray);

 }

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

 String testString = "Whatever";

 *capitalize*(testString);

 **double** testValue = 4.0;

 *square*(testValue);

 **int**[] testArray = {12, 6, 3, 9};

 *sort*(testArray);

 System.***out***.println(testValue); // Third line

 System.***out***.println(testString); // Fourth line

 System.***out***.printf(Arrays.*toString*(testArray)); // Fifth line

 }

}

**First line of output: WHATEVER**

**Second line of output: 16.0**

**Third line of output: 4.0**

**Fourth line of output: Whatever**

**Fifth line of output: [3, 6, 9, 12]**

1. *(3 points each)*Write the output of the following code in the spaces provided.

HINT: Draw a diagram of the contents of both arrays, and track changes.

**import** java.util.Arrays;

**public** **class** FinalExam3

{

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

 // Declare, allocate, initialize array

 **double** doubles[] = {5.0, 4.0, 3.0, 2.0};

 String[][] strings = **new** String[2][3];

 // Modify 1D array

 **for** (**int** i = 0; i < doubles.length; i++)

 doubles[i] = doubles[i] / 2.0 + i;

 // Initialize 2D array

 **for** (**int** row = 0; row < 2; row++)

 **for** (**int** col = 0; col < 3; col++)

 strings[row][col] = "("+ row + "," + col + ")";

 // Print array information

 System.***out***.println(doubles.length);

 System.***out***.println(Arrays.*toString*(doubles));

 System.***out***.println(strings.length);

 System.***out***.println(strings[0].length);

 System.***out***.println(strings[1][2]);

 }

}

**First line of output: 4**

**Second line of output: [2.5, 3.0, 3.5, 4.0]**

**Third line of output: 2**

**Fourth line of output: 3**

**Fifth line of output: (1,2)**

1. *(3 points each)*Write the output of the following code in the spaces provided.

**import** java.io.File;

**import** java.util.Scanner;

**public** **class** FinalExam4 {

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

 **int** i;

 **double** d;

 Scanner scan = **new** Scanner(System.in);

 System.***out***.println(scan.nextLine());

 **char** c0 = scan.next().charAt(0);

 **char** c1 = scan.next().charAt(0);

 System.***out***.println(c0 + "," + c1);

 System.***out***.println(scan.nextInt() + ":" +

 scan.nextDouble());

 d = scan.nextDouble();

 i = scan.nextInt();

 System.***out***.printf("%0.1f, %d\n", d, i);

 scan.close();

 }

}

**Here is what the user types from the keyboard:**

Computer Science

Java Programming

123 567

12.34 4

**First line of output: Computer Science**

**Second line of output: J,P**

**Third line of output: 123:567.0**

**Fourth line of output: 12.3, 4**

1. *(3 points)* Write Java code to instantiate a class **MyClass** with the default constructor into an object called **myObject** and use the object to call a method in the class named **myMethod**, with parameters of type **String**, **int**, and **char** (use any values), and print the return value.

**System.out.println(MyClass.myMethod(“String”, 123, ‘X’));**

1. *(2 points)* Showoneline of Java code that**declares, allocates,**and**initializes**a character array with4 elements with values ‘a’, ‘Z’, $’, and ‘8’, in that order.

**char[] cArray = { ‘a’, ‘Z’, ‘$”, ‘8’};**

1. *(5 points)*Write a method. It should allocate a character array called **cArray** that is the same size as an argument, **sArray**, copy the first character of each element in **sArray** to the corresponding element in **cArray**, and return **cArray**.

**public** **static** **char**[] initials(String sArray[]) {

**char** cArray[] = **new** **char**[sArray.length];

 **for** (**int** i = 0; i < sArray.length; i++) {

 cArray[i] = sArray[i].charAt(0);

 }

 **return** cArray;

 }

1. *(2 points)*Show the **declaration** for a static method called **myMethod** that is public, has an **int** return value, and accepts one parameter which is a two-dimensional array of **doubles**. You do not need to write the code, just show the declaration.

**public int myMethod(double[][] dArray);**