Problem Session 1

# Question 1 - Data Types - 10 points

What do the following Java expressions evaluate to? If an expression does not compile or cause and exception at runtime, put an **X** in both columns.

| Expression | Value | Type |
| --- | --- | --- |
| "1" + 2 \* 6 | “112” | String |
|  |  |  |
| 1 / 2 / 6 | 0 | int |
|  |  |  |
| Integer.parseInt(12) \* 6 | X | X |
|  |  |  |
| (1 >= 26) || (12 >= 6) | true | boolean |
|  |  |  |
| (int) 1.26 / (double) 10 | 0.1 | double |
|  |  |  |
| 1.2e6 % 1.2e5 | 0.0 | double |
|  |  |  |
| 1 <= 2 < 6 | X | X |
|  |  |  |
| Math.min(1.2e6, 1.2e7, 1.2e8) | X | X |
|  |  |  |
| true != false && true > false | X | X |
|  |  |  |
| !!!!!!false | false | boolean |

# Question 2 - Arrays - 10 points

Consider the following Java code fragment.

int[] a = { 1, 6, 5, 3, 0, 2, 4 };  
int n = a.length;  
  
int [] b = new int[n];  
for (int i = 0; i < n; i++)  
 b[a[i]] = i;  
  
int [] c = new int [n];  
for (int i = 0; i < n; i++)  
 c[i] = a[b[i]];

What are the values of the elements in the array b[] and c[] after the above code fragment is executed? Write your answers in the space below.

a[] = { 1, 6, 5, 3, 0, 2, 4 }

b[] = { 4, 0, 5, 3, 6, 2, 1 }

c[] = { 0, 1, 2, 3, 4, 5, 6 }

# Question 3 - Arrays - 10 points

Determine whether each statement below is true or false. Assume no index is out of bounds. Write either “true” or “false” on each line.

**For Part 8-11**:

int[] a = new int [6];  
int [] b = new int [6];  
int[] c = a;  
c[0] = 3;

| Statement | True or False |
| --- | --- |
| 1. Every element in a new array of integers is initialized to 0. | true |
|  |  |
| 3. Every element in a new array of booleans is initialized to true. | false (initialized to false) |
|  |  |
| 4. N-1 is the last valid index into an array of size N. | true |
|  |  |
| 5. 0 is the index of the smallest element in an array of size N. | false (first not smallest) |
|  |  |
| 6. N-3 is the index of the third-to-last element in an array of size N. | true |
|  |  |
| 7. (int) (Math.random() \* N) is a valid index into an array of size N. | true |
|  |  |
| 8. You can create a new array without using the new keyword. | true (int[] c = {1, 2, 3}) |
|  |  |
| 9. Given the array a, b, and c as defined above, a == b evaluates to true. | false |
|  |  |
| 10. Given the array a, b, and c as defined above, a == c evaluates to true. | true |
|  |  |
| 11. Given the array a, b, and c as defined above, a[b[0]] evaluates to 0. | false ( to 3) |
|  |  |
| 12. Given the array a, b, and c as defined above, a[b[c[0]]] evaluates to 0. | false ( to 3) |

# Question 4 - Functions - 10 points

Which of the following statements are true for Java functions (static methods). Mark each statement as either true or false.

| true | false | Statement |
| --- | --- | --- |
|  |  |  |
| true |  | A function can accept more than one argument. |
|  |  |  |
| true |  | You can use double as the return type of a function. |
|  |  |  |
| true |  | A function can call other functions only if those functions are defined in the same .java file. |
|  |  |  |
|  | false | Two functions defined in the same .java file can have the same name only if they have a different number of arguments. |
|  |  |  |
| true |  | If you pass a (reference to an) array as an argument to a function, the function can not only read the array’s entries, but it can also change them. |

# Question 5 - Functions - 10 points

What do each of the following functions return? Fill in the table below.

Assume that the array x, y and z are defined as follows:

int[] x = new int[5];  
int[] y = {1, 2, 6};  
int[] z = {-10, 0, 10};

public static int f(int[] a) {  
 int t = 0;  
 int n = a.length;  
 for (int i = 0; i < n; i++)  
 t += a[i];  
 return t;  
}

| f(x) | f(y) | f(z) |
| --- | --- | --- |
| 0 | 9 | 0 |

public static int g(int[] a) {  
 int n = a.length;  
 for (int i = 0; i < n; i++)  
 a[i] = a[n-i-1];  
 return f(a);  
}

| g(x) | g(y) | g(z) |
| --- | --- | --- |
| 0 | 14 | 20 |

**NOTE**: The arrays x, y and z are reset to their original values before calling h().

public static int h(int[] a) {  
 int n = a.length;  
 for (int i = 0; i < n; i++)  
 a[i] = f(a);  
 return f(a);  
}

| h(x) | h(y) | h(z) |
| --- | --- | --- |
| 0 | 58 | 30 |

# Question 6 - Arrays - 10 points

Which of the following are properties of arrays in Java?

Mark each statement as either true or false by filling in the table below.

| true | false | Statement |
| --- | --- | --- |
| true |  | After you create and initialize an int[] array, you cannot change its length. |
|  |  |  |
|  | false | If a is an array of type char[] and length 10, then a[0.0] is a valid expression that given its first element. |
|  |  |  |
| true |  | It is possible to declare, create and initialize a String[] array without using the keyword new. |
|  |  |  |
|  | false | If a[] and b[] are two different arrays of the same type and length, then the expression (a == b) evaluates to true if the corresponding array elements are equal, and false otherwise. |
|  |  |  |
| true |  | The elements of an array of type int[] are stored in the computer’s memory (i.e., in consecutive memory locations). |
|  |  |  |
| true |  | It is possible to create and initialize a two-dimensional array a[][] of type double[][] such that (a[0].length != a[1].length) |

# Question 7 - Functions, arrays and pass-by-value - 5 points (No Answer is provided)

Consider the following two functions, which purport to negate the values in an integer array:

public static void negate1(int[] a) {  
 for (int i = 0; i < a.length; i++)  
 a[i] = -a[i];  
}  
  
public static int[] negate2(int[] a) {  
 int[] b = new int[a.length];  
 for (int i = 0; i < a.length; i++)  
 b[i] = -a[i];  
 a = b;  
 return b;  
}

Suppose that an integer array a[] in main is declared and initialized to contain the three integers [1, 2, 3]. What will be the contents in the array referenced by a[] after executing each of the following statements? Answer each part independently.

For each statement on the left, write the letter of the best-matching description on the right. You may use each letter once, more than once, or not at all.

| Statement | Description |
| --- | --- |
| \_\_\_\_ negate1(a); | A. [1, 2, 3] |
|  |  |
| \_\_\_\_ negate2(a); | B. [-1, -2, -3] |
|  |  |
| \_\_\_\_ a = negate2(a); | C. [0, 0, 0] |
|  |  |
| \_\_\_\_ negate1(negate2(a)); | D. compile-time error or run-time exception |
|  |  |
| \_\_\_\_ a = negate2(negate2(negate2(a))); |  |

# Question 8 - Properties of types, variables and expressions - 5 points (Answers are not validated)

Which of the following are properties of type, variables and expressions in Java?

Mark each statement as either true or false by filling in the table below.

| true | false | Statement |
| --- | --- | --- |
| true |  | Every variable has a type (such as int, double or String) that is known at compile time. |
|  |  |  |
| true |  | The result of applying one of the arithmetic operators (+, -, \*, /) to two double operands always evaluates to a value of type double (and never produces a run-time exception). |
|  |  |  |
| true |  | If you attempt to use a local variable of type int in an expression before that variable has been assigned a value, Java will substitute the value 0. |
|  |  |  |
| true |  | If a variable is declared and initialized in the body of a for loop, that variable cannot be accessed outside that loop. |
|  |  |  |
| true |  | If you name a variable with all uppercase letters and initialize it to some value, attempting to subsequently change its value would lead to a compile-time error. |