



University of Maryland College Park

Dept of Computer Science

CMSC131 Spring 2011

Midterm I Key

First Name (PRINT): _____

Last Name (PRINT): _____

University ID: _____

Section/TAName: _____

I pledge on my honor that I have not given or received any unauthorized assistance on this examination.

Your signature: _____

Instructions

- This exam is a closed-book and closed-notes exam.
- Total point value is 100 points.
- The exam is a 50 minutes exam.
- For coding problems you do not need to provide pseudocode and you don't need to specify import statements.
- Please use a pencil to complete the exam.
- **WRITE NEATLY.** If we cannot understand your answer, we will not grade it (i.e., 0 credit).

Grader Use Only

#1	Problem 1 (Short Answers)	(36)	
#2	Problem 2 (Java Code)	(22)	
#3	Problem 3 (Java Code)	(24)	
#4	Problem 4 (Java Code)	(18)	
Total		(100)	

Problem 1 (36 pts)

1. (2 pts) How many bits are in a byte?

Answer: 8

2. (2 pts) How many different combinations of 0's and 1's can be represented with 4 bits?

Answer: 16

3. (2 pts) Write a single statement that declares two variables, a and b, and initializes them with the values 7.3 and "Goodbye". (Choose an appropriate type.)

Answer:

`double d = 7.3; or float f = 7.3; (they must have 7.3f, but we will accept 7.3)`
`String message = "Goodbye";`

4. (2 pts) Java programs are "portable" because the Java compiler translates your source code into what?

Answer: bytecode

5. (2 pts) Name two primitive types used to store integers.

Answer: Any two of byte, short, int, long

6. (2 pts) What value would Java compute for the following Java code fragments?

a) `int x = 1; int y = x / 2; // What is the value of y?`

Answer: 0

b) `int w = 17 % 3; // What is the value of w?`

Answer: 2

7. (2 pts) Which of the following could be used to name variables in Java? We're not asking if they are good style, just whether or not they are permissible. (Circle all that apply.)

Gone# element2 10bridges water@basement

Answer: Only one circled is element2

8. (4 pts) Re-write (in the box) the following code fragment using a *for-loop*.

```
int x = 2;
while (x <= 400) {
    System.out.println(x);
    x += 7;
}
```

Answer:

```
for (int x = 2; x <= 400; x += 7)
    System.out.println(x);
```

9. (2 pts) Write the binary representation of 7.

Answer: 111

10. (2 pts) What is an assembler?

Answer: Converts assembly code into machine code

11. (2 pts) What is pseudocode?

Answer: Strategy we used to design compute programs

12. (2 pts) What is **null**?

Answer: Java value representing no address

13. (2 pts) Complete the following assignment so we are able to print the message:

Richard "Rick" Smith

```
String name = "Richard \"Rick\" Smith"
System.out.println(name);
```

Answer: "Richard \"Rick\" Smith"

14. (2 pts) How many distinct String object instances are created in the following code segment?

```
String value = "Baseball";
value = "GoodBye";
String valueTwo = value;
```

Answer: _____

Answer: 2

15. (2 pts) Define an integer constant named MAX_OK_TEMP that has as value 99.

Answer: final int MAX_OK_TEMP = 99;

16. (2 pts) Write the output generated by the following statements.

```
int y = 10;
int x = y++;
System.out.println(y);
System.out.println(x);
```

Answer: 11 10

17. (2 pts) Will the value of x change in the following code? Briefly explain (yes or no answer with no explanation will receive no credit)

```
int x = 20, y = 10;
if ( (y >= 10) || (++x > 20) ) { }
```

Answer: No, due to short circuiting

18. (2 pts) The following code fragment generates an error when run. Why?

```
String k = null;
int x = k.length();
```

Answer: No, because there is no object

Problem 2 (22 pts)

Fill in the method below in order to complete a program called **IdProgram**. The program reads a number of credits (using the Scanner class) and prints the student classification (“A”, “B”, “C”) based on the number of credits associated with a student. Students are classified as follows:

Number of Credits	Type
Less than 40	A
Between 40 (inclusive) and 80 (inclusive)	B
Greater than 80	C

Restrictions/Assumptions

- Use the message “Enter credits” to read the number of credits.
- You must use System.out.println to print the total.
- You should use the Scanner class (Scanner sc = new Scanner(System.in);) to read values.

Answer:

```
public class IdProgram {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter credits");
        int credits = scanner.nextInt();
        String type;
        if (credits < 40) {
            type = "A";
        } else if (credits <= 80) {
            type = "B";
        } else {
            type = "C";
        }
        System.out.println(type);
    }
}
```

Problem 3 (24 pts)

Fill in the method below in order to complete a program called **ReadAge**. The program will keep asking an age value (using the message "Enter age") as long as the user provides a negative value or a value larger than 120. Each time an invalid value is provided, the program will print the message "Invalid value". Once a valid value has been provided, the program will print the provided value using the message "Age: " followed by the value. Use the Scanner class to read the value.

Answer:

```
public class ReadAge {
    public static void main(String[] args) {
        int age;
        Scanner scanner = new Scanner(System.in);
        boolean done = false;
        do {
            System.out.println("Enter age");
            age = scanner.nextInt();
            if (age >= 0 && age <= 120) {
                done = true;
            } else {
                System.out.println("Invalid value");
            }
        } while(!done);
        System.out.println("Age: " + age);
    }
}
```

Problem 4 (18 pts)

Fill in the method below. The method will display the following diagram based on the size value provided in the parameter. For example, for a size value of 4, the diagram will be:

```
*  
**  
***  
****
```

Your solution must handle different size values (not just 4).

```
public static void printDiagram(int size) {
```

Answer:

```
public class Diagram {  
    public static void printDiagram(int size) {  
        for (int row = 1; row <= size; row++) {  
            for (int col = 1; col <= row; col++) {  
                System.out.print("*");  
            }  
            System.out.println();  
        }  
    }  
  
    public static void main(String[] args) {  
        printDiagram(4);  
    }  
}
```