University of Maryland College Park
Dept of Computer Science
CMSC131 Fall 2018
Exam #1 Key

FIRSTNAME, LASTNAME (PRINT IN UPPERCASE):_____

STUDENT ID (e.g. 123456789): _____

Instructions

- Please print your answers and use a pencil.
- Do not remove the staple from the exam. Removing it will interfere with the Gradescope scanning process.
- To make sure Gradescope can recognize your exam, print your name, write your directory id at the bottom of each page, provide answers in the rectangular areas provided, and do not remove any exam pages. Even if you use the provided extra pages for scratch work, they must be returned with the rest of the exam.
- This exam is a closed-book, closed-notes exam, with a duration of 50 minutes and 200 total points.
- You don't need to use meaningful variable names; however, we expect good indentation.

Grader Use Only

#1	Problem #1 (Miscellaneous)	20	
#2	Problem #2 (Military Time)	90	
#3	Problem #3 (Diagram)	90	
Total	Total	200	

Problem #1 (Miscellaneous)

1. (2 pts) Java compilers produce:

   a. Assembly code.
   b. Byte Code.
   c. Machine code (can run on the computer CPU).
   d. None of the above.

   Answer: b.

2. (3 pts) Which of the following could be used to name variables in Java? We are not asking if they are good style, just whether or not they are permissible. Circle all that apply.

   space   while   &horizon   car_component

   Answer: space and car_component

3. (3 pts) Write the binary representation of **19.**

   Answer: 10011

4. (3 pts) Write the decimal representation of the binary number **10111**.

   Answer: 23

5. (3 pts) Write the output generated by the following statements:

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

   Answer: 11 10

6. (6 pts) Answer the two questions below based on the following code where x and m are integer variables.

   ```
   boolean y = ((x > 0) && (++m > 1));
   System.out.println("y: " + y);
   System.out.println("x: " + x);
   System.out.println("m: " + m);
   ```

   a. (3 pts) What is the output when x == 0 and m == 100?

      Answer:

      ```
      y: false
      x: 0
      m: 100
      ```

   b. (3 pts) What is the output when x == 20 and m == 100?

      Answer:

      ```
      y: true
      x: 20
      m: 101
      ```

Problem #2 (Military Time)

Complete the implementation of the program below. The program reads hours, minutes and the string "am" or "pm", and prints the military time associated with that time. To compute military time you multiply the number of hours that have passed since midnight by 100 and add the number of minutes. For example, military time for 5:20 pm will be 17 (17 hours since midnight) * 100 + 20. This will generate 1720. For this program, use the following messages to read data: "hours: ", "minutes: " and "am or pm: ". To display the result use "mil: ". You can assume the user will provide correct values for hours and minutes, however, you need to check whether "am" or "pm" is provided. If something other than "am" or "pm" is provided, the program will not perform any computation.

Below we are providing an example of running the program. Underlined text represents input provided by the user. Remember, your program must work for other values.

One Possible Answer:

```
Scanner scanner = new Scanner(System.in);

System.out.print("hours: ");
int hour = scanner.nextInt();

System.out.print("minutes: ");
int minutes = scanner.nextInt();

System.out.print("am or pm: ");
String am_pm = scanner.next();

if (am_pm.equals("am") || am_pm.equals("pm")) {
        int milTime;
        if (am_pm.equals("pm") && hour < 12) {
                milTime = hour + 12;
        } else {
                milTime = hour;
        }
        milTime = milTime * 100;
        milTime = milTime + minutes;

        System.out.println("mil: " + milTime);
}

scanner.close();
```

Problem #3 (Diagram)

Complete the program below that prints a square. The program will read the size (an integer value) of the square and a character. It will then generate a square with a number of rows and columns that corresponds to size, and where the * character is used for the square border. The provided character will be used for the rest of the diagram. Use the message "Enter size:" and "Enter character: " to read data. You can assume users will provide correct data and a size value larger than or equal to 3. Notice you must read the character as a character and not as a string.

One Possible Answer:

```
Scanner scanner = new Scanner(System.in);
System.out.print("Enter size: ");
int size = scanner.nextInt();
System.out.print("Enter character: ");
char filler = scanner.next().charAt(0), middleChar;

for (int row = 1; row <= size; row++) {
        if (row == 1 || row == size) {
                middleChar = '*';
        } else {
                middleChar = filler;
        }
        System.out.print("*");
        for (int col = 2; col <= size - 1; col++) {
                System.out.print(middleChar);
        }
        System.out.println("*");
}
scanner.close();
```