

CIS 201 Final: Chapters 1 - 8 Fall 2019

Name:

Score:

/90

Directions: This is a closed book, closed notes final. Please place your answers in the space provided. There are 90 points on this examination.

1. [6 pts] Evaluate the following expressions and for each expression in the left-hand column, indicate its value in the right-hand column. List a constant of appropriate type (e.g., 7.0 for a double, true or false for a boolean, Strings in "" quotes).

a. `1 + (2 + 3) + 4 + "(5 + 6)" + 7 + (8/4) * 10` =

b. `4 < 10! = (5 == 6 || 9 >= 9)` =

c. `Math.abs(-7) + 5 % 8 - 11 % 4 % 2` =

2. [6pts] Assuming the following variable has been declared:

```
String str = "Gandalf the GRAY";  
int x = 1234567;  
int y = (x % 100000)/10000;
```

Evaluate the expressions below.

a. `str.toLowerCase().substring(9, 13) + str.substring(3, str.length() - 4)`

Answer:

b. `str.substring(str.indexOf("h")) + str.charAt(str.length() - 10)`

Answer:

3. [5 pts] What output is produced by the following program?

```
public class MysteryTouch {
    public static void main(String[] args) {
        String head = "shoulders";
        String knees = "toes";
        String elbow = "head";
        String eye = "eyes and ears";
        String ear = "eye";

        touch(ear, elbow);
        touch(elbow, ear);
        touch(head, "elbow");
    }

    public static void touch(String elbow, String ear) {
        System.out.println("touch your " + elbow + " to your "
+ ear);
    }
}
```

Answer:

4. [6 pts] Consider the following method:

```
public static void arrayMystery(int[] a) {
    for (int i = 1; i < a.length - 1; i++) {
        a[i] = (a[i - 1] + a[i + 1]) / 2;
    }
}
```

what values would be stored in the array after the method arrayMystery executes for the integer array below is passed as a parameter to it.

```
int[] a2 = {2, 1, 3, 4};
arrayMystery(a2);
```

Answer:

5. [10 pts] Random Numbers and Programming: Write a method called `rolls()` . This method generates random integers between 1 and 50, inclusive, printing out each value that it generates. When a 25 is printed it returns the number of rolls(including the 25) that it took to get the 25. Here is example of a statement that calls this method:
`System.out.println("It took " + rolls() + " rolls to get a 25");`

Here is are three examples of what the execution of the above statement might look like:

8 2 36 78 43 8 25
It took 7 rolls to get a 25

25
It took 1 rolls to get a 25

Answer:

6. [10 pts] Write a static method named `countLastDigits` that accepts an array of integers as a parameter and examines its elements to determine how many end in 0, how many end in 1, how many end in 2 and so on. Your method will return an array of counters. The count of how many elements end in 0 should be stored in its element at index 0, how many of the values end in 1 should be stored in its element at index 1, and so on.

For example, if an array named `list` contains the values {9, 29, 44, 103, 2, 52, 12, 12, 76, 35, 20}, the call of `countLastDigits(list)` should return the array {1, 0, 4, 1, 1, 1, 1, 0, 0, 2} because 1 element ends with 0 (20), no elements end with 1, 4 elements end with 2 (2, 52, 12, and 12), etc.

Answer:

7.[10 pts] Write a method that accepts a String (the String consists of zero or more tokens separated by " " or "\n" or "\t") and writes each token to its own line in a new file named "tokens.txt". If there are no tokens your method should not create or write to a file. If the method receives "john the artist" the new file would contain

```
john
the
artist
```

Answer:

8. [15 pts] Define a class named `TimeSpan`. A `TimeSpan` object stores a span of time in hours and minutes (for example, the time span between 8:00am and 10:30am is 2 hours, 30 minutes). Each `TimeSpan` object should have the following public methods:

a. `TimeSpan(hours, minutes)`

Constructs a `TimeSpan` object storing the given time span of hours and minutes.

b. `getHours()`

Returns the number of hours in this time span.

c. `getMinutes()`

Returns the number of minutes in this time span, between 0 and 59.

d. `add(hours, minutes)`

Adds the given amount of time to the span. For example, (2 hours, 15 min) + (1 hour, 45 min) = (4 hours).

Assume that the parameters are valid: the hours are non-negative, and the minutes are between 0 and 59.

e. `add(timespan)`

Adds the given amount of time (stored as a time span) to the current time span.

f. `getTotalHours()`

Returns the total time in this time span as the real number of hours, such as 9.75 for (9 hours, 45 min).

g. `toString()`

Returns a string representation of the time span of hours and minutes, such as "28h46m".

The minutes should always be reported as being in the range of 0 to 59. That means that you may have to "carry" 60 minutes into a full hour.

Answer

9. [12 pts] A BankAccount class is given below. Write a client program called BankTest that tests all the methods in the BankAccount class.

```
public class BankAccount {
    private double balance; // in dollars
    private String name;
    private String id;
    private int transaction;

    // Constructs a bank account with a given balance, name and account ID.
    public BankAccount(double initialBalance, String theName,
        String theId) {
        balance = initialBalance;
        name = theName;
        id = theId;
    }

    // Constructs a bank account with a zero balance and the given name and
    // account ID.
    public BankAccount(String theName, String theId) {
        this(0, theName, theId);
    }

    // Deposits money into the bank account. Fails if amount is negative.
    public void deposit(double amount) {
        if (amount > 0) {
            balance += amount;
            transaction++;
        }
    }

    // Winthdraws money from the bank account.
    // Fails if amount is negative or greater than the balance.
    public void withdraw(double amount) {
        if (balance - amount >= 0 && amount > 0) {
            balance -= amount;
            transaction++;
        }
    }

    public String toString() {
        return "#" + id + " (" + name + "): $" + balance;
    }
}
```

Answer:

11. [10 pts] Write a nested for-loops to produce the output below. DO NOT write a complete method or class.

```
Odd-----5-----  
Eve----444----  
Odd---33333---  
Eve--2222222--  
Odd-111111111-
```