

CSE 1321L: Programming and Problem Solving I Lab

Lab 5

Selection Structures

What students will learn:

- Logic using selection structures (if/else and switch statements)
- Review of I/O (input and output)
- Review of reading input from the user and storing it into variables
- Review of doing basic calculations with variables to generate a solution

Overview: In this lab, you're going to continue working with conditional statements (if...else..., if, else and switch). These constructs are central to all but the simplest of programs, so you need to become completely fluent in using them to solve problems.

We've now seen if/else statements a few times. However, there's an old "legacy" way of accomplishing the same thing, so we wanted to show it to you (you're **NOT** required to use it). Imagine that we run a restaurant and we need to determine the cost of a meal; kids under 10 eat for free. One approach to coding this would be a simple if/else statement

```
int age = 15; float
costOfMeal;
if (age <= 10) {
    costOfMeal = 0.0f;
}
else {
    costOfMeal = 9.99f;
}
```

However, we can shorten this code significantly by using the "legacy" form that looks like this:

```
int age = 15;
float costOfMeal = (age <= 10? 0.0f: 9.99f);
```

These two pieces of code accomplish the same thing, but we need to explain what's going on. The "legacy" version uses a set of parentheses that have three parts separated by a question mark and a colon. The first part (`age <= 10`) is a conditional statement that resolves to true or false; the second part is a value that is assigned to `costOfMeal` if that expression is true; and the third part is a value that is assigned to `costOfMeal` if the expression is false. The way to remember this is the first part of the expression is asking a question – is age less than or equal to 10?

Again, you **don't** have to use this style of programming, but we wanted you to recognize it when you see it used in examples of other people's code.

As with previous weeks, all labs should have the appropriate class names and file names, Lab5A, Lab5B, and Lab5C (.java, .cs, .cpp). You are not required to use repl.it for this set of labs unless you want more practice with it.

Lab5A: Driving me Crazy!

Please write a program that determines if an applicant is eligible to buy a car; for the sake of argument, we are assuming that you need to have a commercial license (not driving permit) for the applicant to be eligible to purchase a vehicle. An applicant would have not been able to get a commercial driving license without having had a driving permit.

You will need to use nested if-statements to make this program function properly.

First question will ask if the applicant has a permit. Second question should ask if the applicant has a license and should be asked *if and only if* the applicant had answered "Y" or "y" to the first question. If the user answers "N" or "n" to any of the above to questions the applicant should not be eligible for the purchase of a vehicle **and the program should not ask any further questions.**

If and only if the applicant has answered "Y" or "y" to both the questions will they be eligible to purchase a vehicle.

Please refer to the sample outputs below to get an idea what *should be* printed out. The program should read the input answers (**as chars**) and return an appropriate reply as the output. The sample output is shown below. Please note that the input can be **either upper or lower case.**

Remember, the class name should be Lab5A.

The user input is indicated in **bold**.

Sample output #1:

```
Do you have a driving permit (Y/N)? N  
Driving permit is a prerequisite to purchase a vehicle!
```

Sample output #2:

```
Do you have a driving permit (Y/N)? Y  
Do you have a commercial driving license (Y/N)? n  
Commercial driving license is a prerequisite to purchase a vehicle!
```

Sample output #3:

```
Do you have a driving permit (Y/N)? y  
Do you have a commercial driving license (Y/N)? Y  
Congratulations! You can purchase a vehicle, let's start talking options!
```

Intentionally Left Empty, please proceed to Next Page

Lab5B: Sensible Division

Heads up! You will need to use the **modulus function** for the question. If you do not know or have any questions about this function, please ask your Lab Instructor. A brief explanation of the function is presented below.

The modulus function, which looks like this %, is a function that can be used in the Java, C#, and C++ programming languages much like the division /, multiplication *, addition +, and subtraction - functions. This function is like the division function, however, the idea behind it is that it should provide you with the remainder of two divided integers (think long division) rather than with the answer of that division.

For example, if we were to **integer** divide 5 and 2 (`int w = 5 / 2`) the answer of this division would be 2 (remember this is integer division). However, if we were to mod 5 and 2 (`int w = 5 % 2`) then the answer would be 1.

Keeping that in mind, if we were to mod the number that is left of the modulus sign with a number that it is divisible by on the right of the modulus sign, what would the answer be?

For this exercise, please design a program that determines if the number that was entered in as input by the user is divisible by 2, 3, or 5. If it is not divisible by any of these numbers, please just print out "This number is undetermined".

Disclaimer: The answer that should be provided by your program should be the smallest number that equally divides by the number entered in. For example, 200 in sample output #2 is both divisible by 2 and by 5, however, your program should prioritize printing out it is divisible by 2 rather than it being divisible by 5. **Only numbers that are strictly divisible only by 5 should print "This number is divisible by 5" for example like 25.**

Remember, the class name should be Lab5B.

The user input should be an integer and sample runs are shown below. User input is in **bold**.

Note: **You must use a switch statement as part of your solution.**

Sample output #1:

```
Enter an integer: 2
This number is divisible by 2
```

Sample output #2:

```
Enter an integer: 200
This number is divisible by 2
```

Sample output #3:

```
Enter an integer: 13
This number is undetermined
```

Sample output #3:

```
Enter an integer: 25
This number is divisible by 5
```

Lab5C: What's the point?

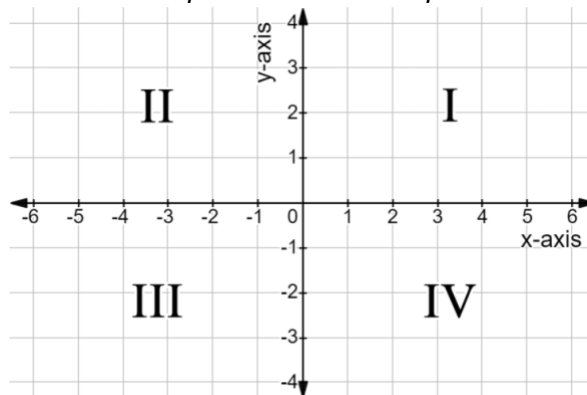
For this lab exercise, you will need to create a program that determines the location of a point on the Cartesian plane. The program should take the x-coordinate and y-coordinate of a point in a Cartesian plane as input. The inputs should be *integer* values.

The program should then print out the entered values and the location of the point on the Cartesian plane.

The possible locations for the point (the outcomes) are:

- On the origin (0, 0)
- On the x-axis
- On the y-axis
- In the first quadrant
- In the second quadrant
- In the third quadrant
- In the fourth quadrant

Example of a Cartesian plane



Remember, the class name should be Lab5C.

Please follow the example of the sample run is shown below. User input is in **bold**.

Sample output #1:

Enter x: **0**

Enter y: **0**

This point is the origin.

Sample output #2:

Enter x: **0**

Enter y: **4**

This point is on the y axis.

Sample output #3:

Enter x: **-5**

Enter y: **0**

This point is on the x axis.

Sample output #4:

Enter x: **1**

Enter y: **1**

This point is in the first quadrant.

Sample output #5:

Enter x: **-1**

Enter y: **-1**

This point is in the third quadrant.

Instructions:

- Programs must be working correctly.
- **Programs must be saved in files with the correct file name.**
- If working in Java or C#, class names must be correct.
- Programs must be working and checked by the end of the designated lab session.
- **Programs (only .java, .cs or .cpp files) must be uploaded to Gradescope by due date.**