

# CSC 1300 LAB 3

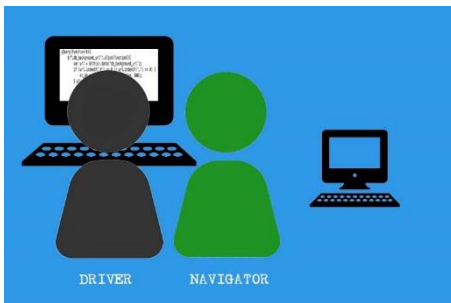
Fall 2024, Written by April Crockett, Last Updated Sept. 11, 2024

## Learning Objectives

- Relational operators & logical operators
- Branching (if statement, if/else statement, if else if else statement)
- Switch statements
- Generating Random Numbers
- Character Data type

## Paired Programming Option

You may complete this lab assignment alone or you have an OPTION to complete this lab with a lab partner using paired programming techniques. If you choose to pair program, follow the directions in this section. Your first step is to **exchange preferred contact information** just in case you are unable to complete the lab during lab and need to meet outside of lab class to finish.



### Submission in iLearn

You will both upload the same exact zip file to your Lab 3 assignment in ilearn. **Each source file should have both of your names in the comment block at the top.** Both students will receive the same feedback and grade.

### How to Pair Program

One of you can start writing (or debugging) the initial code (DRIVER) while the other reviews and suggests improvements (NAVIGATOR). **Take turns regularly (every 10 to 15**

**minutes)** to ensure both of you are actively involved.

## Part 1: Debugging Code with Branches

1. Download the given files from the lab 3 assignment in ilearn, which contains a source file named **lab3a\_given.cpp**.
2. Save this file as **lab3a.cpp** in your **Lab3** folder.
3. This file has many syntax errors and a few logic errors that you must fix. You will need to know how relational expressions with relational operators and logical operators work.
4. Once you have fixed all the syntax errors, run the program and make sure it works with the same output provided in this document.

## Basics of Logical Operators

The AND **&&** means that both relational expressions on either side of the **&&** must be true for the whole expression to be true. The OR **||** means that at least ONE relational expression on either side of the **||** must be true for the whole expression to be true.

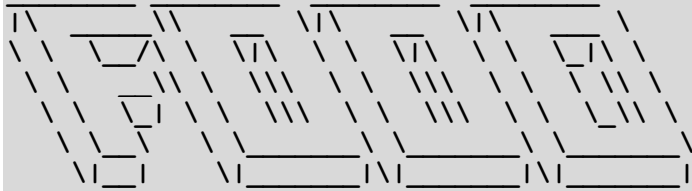
- T && T = T
- T && F = F
- F && T = F
- F && F = F
- T || T = T
- T || F = T
- F || T = T
- F || F = F

## Character Function toupper()

There are several character functions you can use in your code available in the `<cctype>` library. Make sure this library is included in this program, because you will need to use the `toupper()` function, which will take a character and then return the uppercase version of the character. Note that this function can only take in ONE character and only returns ONE character.

## Sample Output

The user's input is highlighted in yellow.



What is your top three favorite foods?

FOOD 1: pizza

FOOD 2: spaghetti

FOOD 3: chips & salsa

Spaghetti is one of my favorite foods, too!

Here are your favorite foods in alphabetical order:

Chips & salsa

Pizza

Spaghetti

## Part 2: Pig is Going on a Picnic



Your **pig** is excited to have a picnic at the lake in the mountains! As it searches for food, it stumbles upon a random food item. It might be something delicious like an **apple** or **corn**, or something terrible like a **pile of mud**. The pig must decide whether to eat the food or leave it. But choose wisely, because eating bad food might spoil the picnic!

**Food Generation:** Use a **random number generator** to create one of three possible food items (below). Use the **rand()** function to generate a random number between 1 and 3. Don't forget to seed the random number generator!!

1 (apple) = healthy food

2 (mud) = bad food

3 (corn) = healthy food

**User's Decision:** After the random food item is generated, the user will decide whether the pig should eat it or leave it.

Use a **char** variable to capture the player's input:

e = eat the food.

l = leave the food.

If the user enters in any other letter other than e or l, the output should say something like "Invalid choice! The pig decided to walk away." You may assume the user will enter in a single character.

**Results:**

If the pig eats healthy food (apple or corn), display a positive message.

If the pig eats bad food (mud), display a negative message.

If the player chooses to leave the food, display a neutral message.

## Sample Outputs

---

User input is highlighted in **yellow**.

### Sample Output 1

Your pig is going on a picnic in the mountains!

The pig found an apple! Should it eat it (e) or leave it (l)? **e**

Yum! The pig enjoyed the food.

### Sample Output 2

Your pig is going on a picnic in the mountains!

The pig found some mud! Should it eat it (e) or leave it (l)? **e**

Yuck! The pig got sick from eating the mud.

### Sample Output 3

Your pig is going on a picnic in the mountains!

The pig found an apple! Should it eat it (e) or leave it (l)? **p**

Invalid choice! The pig decided to walk away.

### Sample Output 4

Your pig is going on a picnic in the mountains!

The pig found some mud! Should it eat it (e) or leave it (l)? **l**

The pig decided to leave the food and continue the picnic.

### Sample Output 5

Your pig is going on a picnic in the mountains!

The pig found some corn! Should it eat it (e) or leave it (l)? **e**

Yum! The pig enjoyed the food.

## Sample Output 6

Your pig is going on a picnic in the mountains!

The pig found an apple! Should it eat it (e) or leave it (l)? **l**

The pig decided to leave the food and continue the picnic.

## Part 3: Switch Statements

The **switch** statement is another conditional statement that can be used in place of if / else-if / else statements when the statement uses the value of an integer expression (or a char) to determine which group of statement(s) to branch through.



## DIRECTIONS

- Retrieve the given file **lab3b\_given.cpp** and save it as **lab3b.cpp** in your Lab3 folder.
- Compile and run the program multiple times to see the different output you get with different inputs. Try each letter grade and try putting in something like "P".
- Put BOTH your name and your partner's name and date in the comment block.
- Rewrite this program to use a **switch** statement instead of **if and else if** statements.
- The output should be identical before and after you rewrite the program.

## Part 4: Fill Out the Lab Report

You will fill out this lab report for every lab and it is part of your grade. To get credit, you must upload a screenshot of the confirmation page to this lab assignment. Name your screenshot **lab3ReportProof**.

**Lab Report Link:** [https://tntech.co1.qualtrics.com/jfe/form/SV\\_d6BGc6kzQdSvBmS](https://tntech.co1.qualtrics.com/jfe/form/SV_d6BGc6kzQdSvBmS)

## What to Turn In

Create a zip file named **labPartner1username\_labPartner2username\_lab3** containing the following .cpp files and upload it to ilearn. Replace labPartner1username with one lab partner's TTU username and replace labPartner2username with the other lab partner's TTU username. Example: **jdean42\_acrockett43\_lab3.zip**

- **lab3a.cpp**
- **lab3b.cpp**
- **lab3c.cpp**
- **lab3ReportProof**

Remember, both lab partners should upload this zip file to their ilearn assignment.