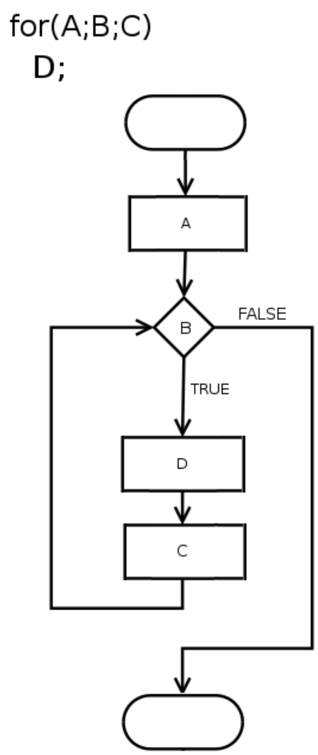
Conditionals

Without conditionals, we cannot decide anything as decisions are everywhere. Conditionals are a method of taking different actions based on different conditions. Conditions mean whether something happens or does not happen. For example, I could do my homework or I cannot do my homework. This is a condition. I have choices. The idea of conditionals in this example is that I take a different action, such as eating, when I have done the homework and a different action, such as doing the homework, when I have not done the homework.



Reference: https://en.wikipedia.org/wiki/Flowchart

In computer science, conditionals have two special features: inputs and outputs. Inputs are what you put into a “special box” and outputs are what you get out of this “special box.” This “special box” is an algorithm. A conditional is a simple example of an algorithm. The inputs to my example algorithm before is whether I have done the homework or I have not done the homework. The outputs are eating food or doing my homework. In computer science, the output that is performed is said to be “returned.” In the example, the output of eating food is “returned” when I have already done the homework. Conditionals sure are tasty! Now, let’s have fun with some exercises!

Exercise 1

Objective: Understand the format of a conditional structure

Assume that we are in a house with a big refrigerator with some food in it. Our goal is to eat, drink, and/or do nothing based on certain conditions. Consider the following sequence of steps:

1. Open the refrigerator.
2. Check for eggs in the refrigerator
3. If there are eggs, make an omelet.
4. If not, check for lemons in the same refrigerator.
5. If there are lemons, make lemonade.
6. If not, close the refrigerator.

This sequence of steps can be done by not only a human, but a computer as well! A computer also does it much faster. However, computers do not understand a human language all the time. For this reason, we have specific languages for the computer to understand. The same sequence of steps can be expressed in one simple way to the computer as follows:

```
open refrigerator

if eggs in refrigerator:
    make_omelet
else:
    if lemons in refrigerator:
        make_lemonade
    else:
        close refrigerator
```

Many features and grammar are not important to understand here. There are many rules in the language that a computer understands. However, they are sometimes similar to human languages. The sequence of steps is also in order just like how a human would read it as! This is because the computer does what the humans tell the computer to do!

a) This procedure involves assessing environmental conditions. What are the conditions that are evaluated during the procedure?

Possible answer:

The conditions that are evaluated are checking for eggs in the refrigerator and potentially checking for lemons in the refrigerator. These can be identified by the word “if” and through the recognition that these are conditions that either happen or do not happen.

b) Is it possible that, at the end of this process, we have both omelet and lemonade? (*Hint*: follow the sequence of steps in order and see the relationship between the steps!!)

Possible answer:

No. If we follow the process step by step, we realize that we have a large first condition of checking whether there are eggs or not in the refrigerator. If there are eggs, then we simply make omelets. We do not go into the “else” condition because we did find the eggs in the refrigerator. Therefore, the process stops. If we did not find the eggs in the refrigerator, then we do go into the “else” condition. This means that we will now have another check for whether there are lemons in the refrigerator or not. If there are lemons in the refrigerator, then we make lemonade. If there are no lemons in the refrigerator, then we simply close the refrigerator and the process stops. With these steps, we have some options for stopping the process: 1) We make an omelet and stop the process 2) We make lemonade and stop the process and 3) We close the refrigerator and stop the process. We cannot have both omelet and lemonade.

c) Rewrite the sequence of steps so that it is possible to have only omelets, only lemonade, both or none, depending on the availability of ingredients in the refrigerator. In this challenge, it is only possible to check if there is one ingredient at a time in the refrigerator. (*Hint*: take this challenge one step at a time!)

Possible answer:

We can simply have conditions that we evaluate one at a time. First, we simply check if there are eggs in the refrigerator. If there are eggs, then we make omelet and move on. If there are no eggs, then we simply move on. Next, we check if there are any lemons in the refrigerator. If there are lemons, then we make the lemonade and move on. If there are no lemons, then we simply move on. At the end, we close the refrigerator. This sequence of steps can mean that we have many options for the output: 1) We have eggs, but not lemons, in the refrigerator and we make omelet only 2) We have lemons, but not eggs, in the refrigerator and we make lemonade only 3) We have both eggs and lemons in the refrigerator and we make both omelet and lemonade, and 4) We do not have either eggs or lemons in the refrigerator and we make neither lemonade or omelet.

These steps can be written as the following:

1. Open the refrigerator.
2. Check for eggs in the refrigerator

3. If there are eggs, make an omelet and move to step 5
4. If there are no eggs, move to step 5
5. Check for lemons in the refrigerator
6. If there are lemons, make lemonade and move to step 7
6. If there are no lemons, move to step 7
7. Close the refrigerator

d) (optional- challenging!) Can you try writing your sequence of steps from part (c) in a way that a computer might be more likely to understand? Use the example given in this exercise as a way to structure your answer.

Possible answer:

The same sequence of steps can be expressed in one simple way to the computer as follows:

```
open refrigerator

if eggs in refrigerator:
    make_omelet

if lemons in refrigerator:
    make_lemonade

close refrigerator
```

The important aspect to note is that the program will be read from the top to the bottom and from the outside to inside. In other words, we will first open the refrigerator no matter what. Then, we will do the check for the eggs in the refrigerator. If there are eggs, then we will make the omelet. However, we will now move on to the next line in the sequence of steps, also known as a computer program. Now, we check to see if there are lemons in the refrigerator. If there are, we make lemonade and continue onwards with the program.

Exercise 2

For each of the following problems, state the conditions that should be evaluated to solve the problem. Note that some problems are not conditional problems.

a) The problem of determining whether or not the current year is a leap year or not.

Possible answer:

In order to determine whether or not the current year is a leap year or not, we need to first know the current year. In programming, this can be stored in what is called a variable. Then, we need to check a couple of conditions for what makes a year a leap year. A leap year is defined as a year with an additional day. However, this is not easy to translate in terms of a language/condition that a computer can easily know about a given year. Therefore, we need to do a check given some assumptions. One way to do this would be to correctly assume that the year 2020 is a leap year and that leap year happens every 4 years. This would allow us to check if the given year is a leap year by checking to see if the given current year is 2020 or another year that is within some multiple of 4 away from 2020. If this is true, such as the year 2028, then the current year is a leap year. If this is not true, then the current year is not a leap year.

b) The problem of printing out a number to the computer screen if the number exists.

Possible answer:

This is not a problem that needs to be solved from using conditionals because the output does not depend on some potential different choices of input actions. In other words, we always print the number to the computer screen assuming that we have a number that exists. There are no other choices, such as only printing the number when the number is even. This would require conditionals, but since we don't have that, this problem does not need conditionals.

c) The problem of allocating vaccines for COVID-19 to the correct priority group of people. Assume a correct priority group here is assigned as elderly people over the age of 65 with some previous medical conditions.

Possible answer:

We would need to check certain characteristics of what makes a priority group. In this case, we would have a condition that checks for whether the person is over the age of 65 or not. In addition, we will also have a true or false value condition that checks whether the person has previous medical conditions or not. If both of these conditions are true, then we can allocate more vaccines for this group. If one or both of these conditions are false, then we do not allocate vaccines to this group.

d) The problem of spending more money based on the country's current budget. Assume there are three characterizations of the budget: accurate, too low, or too high.

Possible answer:

In this case, we would need to know the country's current budget in order to check whether this budget is accurate, too high, or too low. Depending on this condition, we can take different actions. For example, if the budget is accurate or too low, we do not spend more money. If the budget is too high, then we can spend more money since the financial situation is strong.