



APCSP Vocab + Concepts

Vocabulary:

Abstraction:

The process of hiding the complexity or detail of an operation while still presenting its essential characteristics. Putting the many instructions that may carry out a given operation into a method with a name that represents the operation is an example of abstraction.

Call a Method:

The place in a program where you tell the computer to go execute a method (and it jumps to where that method is defined and then comes back).

Execute/Play:

Tell the computer to execute instructions you have given it.

Method:

A named collection of instructions that naturally belong together to do something/task.

Method Definition:

The place where you describe/put the instructions that should be executed when this method is called

Parameters:

Have a single piece of code (the method body) that works on a different actor or character every time it is called - depending on the character that is passed into the method when it is called.

Parameter Type:

This is specified at the time the parameter is created, and indicates what kind of value to expect when the method is called - for example, will it be a character in the world or a numerical value?

Default Value:

If a parameter is added to an existing method, then Alice (Programming UI example) automatically updates every call to the method to include a parameter value. Alice uses a default or standard value for this - for example, for the numeric type, Alice uses the number one.

Event:

Something that happens externally to your program, often initiated by your audience - the users of your programs.

Event Handler:

This is a connection that you make between one or more particular events and code (a method) you have written in your program. The connection and the code

together enable your program to respond to the events occurring outside your program

Key Press Events:

These are events that occur as a result of the user pressing keys on the computer's keyboard.

Mouse Press Events:

These are events that occur as a result of the user manipulating the mouse - there are events for pressing and releasing the buttons on the mouse, as well as for whenever the mouse moves.

Function:

A function calculates a value. We can use functions to hide away complex mathematical expressions – a form of abstraction. When we execute a purple function tile (usually when we execute a method where the function is used as a parameter), we say that we "call the function" (just like calling a method). However, when a function is done executing it has calculated a value – and the value essentially replaces the function call.

Mathematical Expression:

A combination of numbers and mathematical operators that give a numerical result (later we will see it can also produce a logical result). $(2 + 5)$ and $(\text{length} * \text{width} * \text{height})$ are examples of mathematical expressions.

Return (from a function):

When functions are executed they calculate a value which we say is then returned to the place where the function was called. In essence, the value calculated replaces the name of the function call. When you reach a place in a code where there is a return it flies back to where the function was called

returning the value calculated. If there is any code after the return, it will not be executed (we will see examples of this in later modules).

Conditional Expression:

Conditional Expressions are another name for If/Else statements. These expressions evaluate a function (for instance – equal to, greater than, less than, etc.) and depending on the result, perform a certain action. That is, they perform an action on the condition to determine if it is "true".

Conditional Execution:

Conditional Execution is a term that describes actions resulting from Conditional Expressions. Based on the conditional expression, one piece of code is executed, another is not.

If/Else:

The If/Else tile is Alice's version of Conditional Expressions. These green colored tiles say: If something, then do A. Else (otherwise), do B.

Boolean Operators:

Boolean operators are terms that are used to logically evaluate statements. Alice uses the Boolean operators And, Or, and Not. Boolean operators return a value of either "true" or "false".

Both/And Operator:

A mathematical operator (like "+" or "/") that operates on two Boolean values. If both values are true, then the expression is true. If either one or both of the values are false, then the expression is false.

Compound Boolean Expression:

An expression which evaluates to true or false which is made up of multiple Boolean expressions combined with and and/or or.

Nested if/else statements:

When one if/else statement has inside either it's "true" section of code or it's "else/false" section of code another if/else statement that enables another condition to be checked to control execution.

Either/Or/orBoth operator:

A mathematical operator (like "+" or "/") that operates on two Boolean values. If either one of the values OR both values are true, then the expression is true. If both of the values are false, then the expression is false.

Counted loop:

A control structure (tile) that causes the instructions listed on the tile to be executed some number of times. This number can be a constant number (like "4" or "10") or it can be a value calculated by a function when the program runs (bird's distance to tree).

Nested loops:

A loop is said to be nested inside another loop when one loop tile is on top of another loop tile. This can be done with both counted and while loops, but it is more common to have nested counted loops. The loop tile on top of the other is called the Inner Loop and the base loop is called the Outer Loop.

While loop:

A control structure (tile) that causes the instructions listed on the tile to be executed some number of times -- which is not known or fixed beforehand. Instead, a boolean expression is used to control whether the instructions should be executed (again). If the boolean expression is true, then the computer executes the code on the tile -- then control goes back to the top of the tile and the boolean expression is checked again to see if the code on the tile should be repeated. It is possible for the code on a while loop tile to never be executed if the boolean expression evaluates to false when the loop tile is encountered.

Lists:

A list of objects allows objects to be grouped together so that you can perform the same action on all the items in a list, or so that you can have access to certain objects that have similar properties.

Randomness:

We can choose random numbers using a random number generator. Random numbers are useful for making objects do an action with variance, and can be used for choosing random objects too.

Concepts:

Copyright laws/Creative Commons:

Look this one up on the Internet. It defines how creators of online media are protected. Basically: don't plagiarize! Cite where you get work/images/videos!

High level programming & low level programming:

High level: The programming language you use is as close to the English language as possible. For example, Python and Java are high level programming languages.

Low level: The programming language computers use (all the 1's and 0's).

For-Loop/While-loop/nested for-loop:

Know the differences between these and when to use them!

Hint 1: For loops are used when you **know** how many times you want your code to run.

Hint 2: While loops are used when you **don't know** how many times you want your code to run but you want it to run **until a certain event happens**.

If-else statements:

Read if-else statements carefully! Only one event will happen in an if-else statement. If the "if" statement **does not** occur, the else statement **must occur**. Whereas if the "if" statement **does** occur, the else statement **will not** occur.

Arrays:

Arrays on the APCSP exam are generic. In most programming languages, arrays start at index 0, however **for the APCSP exam, indices start at 1**.

Turtles:

You will be given an algorithm that makes the Turtle (It looks like a triangle. The longest point of the triangle indicates which direction it is facing) move and you need to choose the correct map that makes the Turtle reach the finish line.

An Example is provided below:

Algorithm:

move_forward()

move_forward()

rotate_left()

move_forward()

move_forward()

rotate_right()

move_forward()

rotate_left()

ChoiceA: (hint: The turtle is facing North/Up)

			END
▲			

ChoiceB: (hint: The turtle is facing South/Down)

▼			
		END	

Booleans:

A boolean is a conditional statement. Depending on if the condition is met (true), code will run. Some examples include if statement conditions or loop conditions. Ex. if $x > 0$, while $x < 10$, etc.