**Task 2.6**

access modifier

Determines the visibility/accessibility of a class or member.

accessor method

A simple method used to define a public interface to read/query the value of private instance variable/s.

algorithms

A sequence of instructions/statements that collectively perform computational function on some data.

argument list

The list/group of parameters/values that are either input to or output from a method.

assembly

For the purposes of this unit, a single executable, library (DLL), or other compiled resource.

breakpoint (debugger concept)

A nominated statement where the debugger will interrupt the execution of the program to allow .

call stack (debugger concept)

Records the method calls that lead to the current .

composite formatting

Used to prepare textual strings/output that allows data to be placed/formatted in fixed width fields and using pre-defined layouts (such as currency formats).

compound operator

An operator that combines the functionality of the assignment and an arithmetic operator, e.g., x += y is equivalent to x = x + y.

condition

An expression of a test to be applied against the state of the system, usually involving either a equality operator or a relational operator.

conditional AND

An operator (&&) used to combine two conditions such that both conditions are required to be true for the combined condition to be true.

conditional OR

An operator (||) used to combine two conditions such that either condition can be true for the combined condition to be true.

continue (debugger concept)

To continue the execution of the progrma under a debugger until the next breakpoint.

dangling else

Where two if/if-else statements have been written such that the interpretation of the two is ambiguous. In particular it is unclear whether an else keyword and statement belong to which of several if statements.

decrement

The operation to subtract the value of 1 from a variable.

enumerated data type

A type of data that is defined by the programmer that uses symbolic names (or textual names) for the values, e.g., defining a type Month with values January, February, etc.

equality operator

Two possible operators that can be used in a condition that can either test for equality (==) or inequality (!=).

increment

The operation to add the value of 1 to a variable.

instance method

A method that is associated with an instance (object), i.e., the method conducts functionality that queries or modifies the instance variables of that object.

instance variable

A variable that is associated with an instance (object), i.e., there is one variable per object that exists.

iteration

Two definitions: (i) a term used to generally refer to repetition control structures, and (ii) a term used to refer to a single execution of statements of the loop body.

logical negation

An operator (!) used to reverse the boolean value/outcome of a particular condition (!).

logical operator

Operators used for combining and modifying the outcome of a condition, including the conditional AND, conditional OR, and the logical negation operators.

loop body

The statements that form the executable part of a loop that are performed once for each iteration of the loop.

mutator method

A simple method used to define a public interface to write/modify the value of private instance variable/s.

property

A programming language construct that is often found in modern object-oriented programming languages that replaces accessor and mutator methods with something more intuitive.

reference type

A type of variable which stores a memory address, thus any data access requires two memory accesses: one to read the address of the data and a second to read the data itself. - in C#, all custom types defined by a class are reference types.

relational operator

Operators that can be used in a condition that test how one value compares to another, including less than(<), less="" than="" or="" equal="" to="" greater="">), and greater than or equal to (>=).

repetition control structure

Defined under the structure theorem, a control structure which provides the ability to execute one or more statements some number of times, i.e., to repetitively execute statements..

return value

A single value that usually indicates the overall result/outcome of a method. The return value can also return data generated as part of the methods operation, although in modern programming languages output parameters are often used for this purpose.

run/start (debugger concept)

To begin the execution of the program under a debugger until the next breakpoint.

selection control structure

Defined under the structure theorem, a control structure which provides the ability to decide which statements to execute depending on a conditional test on the state of the system, i.e., to selectively execute statements.

short-circuit evaluation

A property of the conditional AND and conditional OR operators such that if the first condition joined by a conditional AND is false, or if the first condition joined by a conditional OR is true, the remaining conditions will not be evaluated further.

step into (debugger concept)

Executes a single statement, following the execution of the program inside any method calls that are encountered..

step out (debugger concept)

Completes the execution of the current method in a single step, breaking execution either at the return statement for the current method or at the next statement encountered in the calling method.

step over (debugger concept)

Executes a single statement, completing any method invocations in a single step (not following execution inside the method).

structure theorem

A fundamental concept of software development, a proved idea that any algorithm can be expressed using a combination of three control structures: sequence, selection, and repetition..

truth tables

A table containing boolean values that demonstrates the outcome of all possible true/false input states into a conditional expression.

value type

A type of variable which refers directly to data stored in memory - in C#, all simple types and user-defined structures (structs) are value types.

watch variable (debugger concept)

A variable that is constantly monitored/displayed by a debugger during the execution of a program so that a developer can monitor its changes. Breakpoints can often be linked to the value of a watch variable, such that if an attempt is made to change the watch variable or set the variable to a particular value the program will be immediately interrupted for debugging.