ASSIGMENT 3 – PROGRAMMING 1

1128 3519 – ADADEY MICHAEL

1. **Variable declaration and initialization**: are fundamental concepts in programming that involve creating and assigning values to variables.
   In programming, a variable declaration is the process of introducing a new variable to the program. It involves specifying the variable's name and its data type. The data type determines the kind of values the variable can hold, such as numbers, text, or Boolean values. Declaring variables is necessary before they can be used in the program.
   Here is an example of variable declaration in Python:
   age = 0
   name = ""
   salary = 0.0

   Variable Initialization:
   Variable initialization is the process of assigning an initial value to a declared variable. An example is; python
   age = 25
   name = "John"
   salary = 5000.0

2. **Decision statements**: also known as conditional statements, are constructs in programming that allow the program to make decisions and execute different blocks of code based on certain conditions. Commonly used decision statements include "if" statements. The "if" statement is used to execute a block of code only if a specified condition is true. If the condition is false, the code block is skipped. An example in python may look like;
   age = 18

   ```
   if age >= 18:
      print("You are an adult.")
   ```

3. **Loop statements**: are essential constructs in programming that allow repetitive execution of a block of code. They provide a way to efficiently perform tasks that require repeated actions or processing of data. There are typically three types of loop statements: "for" loops, "while" loops, and "do-while" loops. Each type offers different approaches to control the iteration process.
   (i) "for" loop:
   The "for" loop is commonly used when the number of iterations is known or when iterating over a specific range of values. It consists of an initialization statement, a condition for continuation, an iteration statement, and a code block.

   Example in Python:

   ```
   for i in range(1, 6):
   ```

```
print("Count:", i)
```

(ii) "while" loop:

The "while" loop is used when the number of iterations is unknown, and the loop continues as long as a specified condition remains true. It checks the condition before each iteration and exits when the condition becomes false.

Example in Java:

```
java
int i = 1;
while (i <= 5) {
   System.out.println("Count: " + i);
   i++;
}
```

(iii) "do-while" loop:

The "do-while" loop is similar to the "while" loop but guarantees the execution of the code block at least once before evaluating the condition. It checks the condition after each iteration and continues as long as the condition remains true.

Example in C#:

```
csharp
int i = 1;
do {
   Console.WriteLine("Count: " + i);
   i++;
} while (i <= 5);
```

4. Functions/methods are essential building blocks in programming that execute a set of instructions or code statements. They provide a way to organize and modularize code, making programs more manageable and efficient.
Functions/methods enable you to break down complex tasks into smaller, manageable pieces of code. They can accept input parameters, perform specific operations, and optionally return a value as a result. Here is an example of a function/method in python

```
def greet(name):
   print("Hello, " + name + "!")

# Calling the function
greet("John")
```

This is a function declaration, and to do this you need to declare it with a name, specify the input parameters (if any), and define the code block that represents the functionality of the function.