Hi, welcome to Basic Program Structure.

☼ Repetition, that is, Looping. Programmers often refer to any repeating construct as a “loop”, because a diagram of the construct looks like a loop.

We already know that the branching constructs such as if-else statement enable us to make selection. Loop lets us execute a group of statements repeatedly for any number of times.

☼After this lesson, you should be able to :

Explain the need for repetition in programming

Familiarize with the structure of a loop

Apply the loop design strategy when developing loops

Distinguish between Counter-controlled loop and Sentinel-Controlled loop

☼The ability to execute instructions –especially decisions-over and over is the source of considerable power in computing. Most programming languages provide loop construct. We will introduce the structure of a loop, the loop design strategy, Counter-controlled loop, and Sentinel-Controlled loop in this lesson.

☼Suppose that you need to calculate the distances between your home to N number of coffee shops,

If we only have a sequence, We need to repeat certain instruction(s),

e.g., calculate the travel distance between two points,

(highlight) Read horizonDist

Read vertDist

dist = horizonDist + vertDist

Display distance traveled between these two points

exactly *N* times, where ***N*** is the total number of coffee shops. let’s say N equals = 100. We need repeat a hundred times, it is really tedious.

☼Secondly, the program is not reusable. E.g. Different users have different numbers of coffee shops nearby. Hence, the number to repeat the certain instruction has to be changed for different users.

☼So, how do you solve this problem? Most programming languages provide a powerful construct called a loop that a computer program can

Dynamically choose how many times it repeats specific instruction(s) during the program runtime.

that is, program instructions can be repeated dynamically; Sometimes three times, sometimes 1000 times, or sometimes not even repeat at all

☼ The program is the same for different users (we do not need to change it for different numbers of repetitions). The more refined Pseudocode is given as an example.

☼The general structure of a loop contains four basic steps

Initialize: This defines and initializes the loop control variable, which is used to control the number of repetitions of the loop.

Test that is loop-continuation-condition. This evaluates the test condition. If the test condition is true, then the loop body is executed. Otherwise, the loop is terminated.

Loop body/ Statements. The loop body is executed if the condition is evaluated to be true. It contains the actual action to be carried out.

Update. The test condition typically involves the loop control variable that should be modified each time through the execution of the loop body. The loop control variable will go through the test condition to determine whether to repeat the loop body .

However, sometimes a loop may not have all of them.

E.g. Infinite loop is a loop statement that executes infinitely . Why? Because its Test condition is always true,

a one-time execution of a loop body is referred to as an **iteration** of the loop.

☼Let's look at one more example. Computing the average height of the students in a class.

☼ Pseudocode of the program to compute the average height of N students is given here.

Initialize the loop control variable counter to be 0.

Test condition, if counter less than N is true, enter loop body

which is **READ height // Loop body**

ADD height TO sum

Update the loop control variable by increasing the counter by 1

Test again, if it is true, enter the loop body, execute statements in the loop body, increase the counter by 1

Repeat the same group of actions again and again, until counter equals to N, the loop is terminated. Then execute statements after loop statements, which are **COMPUTE average = sum/ counter**

PRINT average

☼Writing a correct loop is not an easy task for novice programmers. Consider three steps when writing a loop.

Step1: Identify the statements that need to be repeated.

Step2: Wrap these statements in a loop like this.

```
while (true) {
        Statements;
}
```

Step 3: Code the loop-continuation -condition and add appropriate statements for controlling the loop.

```
while (loop-continuation-condition) {
        Statements;
        Additional statements for controlling the loop;
}
```

Note that In Designing loops, you need to consider both the loop control structure and the loop body.

☼Basically, there are two types of loops: Counter-controlled loops and Sentinel-controlled loops.

In a Counter-controlled loop, the loop body is repeated for a specified number of times, and the number of repetitions is known before the loop begins execution.

For example, the number of tourists to board the bus is known before any individual really get onto the bus. The tour guide knows the number of group members in his group. So, the loop body" boarding" statement will be executed N times, which is preset.

In a Sentinel -controlled loop, the number of repetitions is not known before the loop begins execution. A loop control variable is typically used to determine the number of repetitions.

An example of a sentinel value is a user input value such as -1, which should be different from regular data entered by the user.

Another example:for the famous chicken rice in Singapore, the sentinel value of stop queuing is the number of chicken hang there becomes 0 and no more replenishment . The total number of customers who execute "queue and buy" loop body is unknown beforehand.

☼When we develop loops, how to determine it should be a Counter-controlled loop and Sentinel-Controlled loop? Let's revisit the example of computing the average height of the students in a class. Clearly, we know the number of students before we start the loop. So, it is a Counter-controlled loop. We modify the pseudocode to emphasize that the iteration will repeat for exactly N times.

☼Let's change the condition a bit. Walk out of the classroom and Compute the average height of people entering Canteen A in a day.

Are we able to know **beforehand** the number of people entering canteen A before we start the loop body? No.

☼Then, how to solve the problem? We need to choose a proper sentinel value as the loop control variable. A sentinel value is a special value that

signifies the end of the loop. We get the height of the students who enter canteen A until it closes.

So time is a good choice.

Loop-continuation-condition is time is earlier than closing time.

☼Quick summary:

The General structure of a loop contains four basic parts

Initialize, Test/ loop-continuation-condition, Loop body, Update of control variable.

The part of the loop that contains the statements to be repeated is called the loop body.

In Designing loops, you need to consider both the loop control structure and the loop body.

Infinite loop is a loop statement that executes infinitely.

A one-time execution of a loop body is referred to as an **iteration** of the loop.

Basically, there are two types of loops: Counter-controlled loops and Sentinel-controlled loops.

In a Counter-controlled loop, the loop body is repeated for a specified number of times, and the number of repetitions is known before the loop begins execution.

In a Sentinel-controlled loop, the number of repetitions is not known before the loop begins execution.

A sentinel value is a special value that signifies the end of the loop.

I hope you enjoy this lecture.

And, I will see you next time.