1.1. Meaning

- Loop is a language that precisely captures primitive recursive functions. The only operations supported in the language are assignments, addition, and looping a number of times that is fixed before loop execution starts.

1.2 Syntax and explain the process flow

- Syntax (init; condition; increment) {
  statement(s);
  }

- The structure of a Loop can be virtually divided into two parts, namely the control statement, and the body. The control statement of a Loop comprises the conditions that have to be met for the execution of the body of the loop. For every iteration of the loop, the conditions in the control statements have to be true. The body of a Loop comprises the block of code or the sequence of logical statements that are to be executed multiple times. There

1.3 - Example while loop

```
#include <stdio.h>
int main()
{
    int i= 1;
    while (i <= 5)
    {
        printf("%d\n", i);
        ++i;
    }
    return 0;
}
```

- Example of do while loop

```
# include <stdio.h>
int main()
{
    double number, sum = 0;
    do {
        printf("Enter a number: ");
        scanf("%lf", &number);
        sum += number;
    }
    while (number != 0.0);
    printf("sum = %.2lf", sum);
    return 0;
}
```

- Example of for loop

```
# include <stdio.h>
int main()
{   int a;
    for (a = 10; a < 20; a = a + 1)
        { printf("value of a: %d\n", a);
        }
        return 0;
}
```

2. The main difference between a while loop and do while loop is that while loop check condition before iteration of the loop. On the other hand, the do while loop verifies the condition after the execution of the statements inside the loop. Furthermore, the while loop is known as the entry-controlled loop.

3. The significance of updating the loop control variable in while statement is that it helps to terminate the loop.

4. The one-token statements continue and break may be used within loops to alter control flow; continue causes the next iteration of the loop to run immediately, whereas break terminates the loop and causes execution to resume after the loop. Both control structures must appear in loops. The break statement is used to terminate the loop immediately. The continue statement is used to skip the current iteration of the loop.