Functions

Program redundancy can be reduced if we have predefined statements for repeated used operations, known as function

A Function

```
return_type  function_name ( parameter list)
{
// body of the function
}
```

Function definition: It consists of

(i) a function header

(ii) a function body

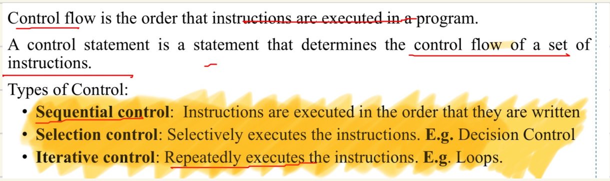
Function declaration: It tells the compiler about the function name, its return type and parameters

formal arguments: declared in function definition ↙ They act as local variables that are created upon entry and destroyed upon exit.

actual arguments: declared and passed during function call

Function Prototype

a declaration in computer programming that specifies a function's name, type signature, and parameters, but omits the function body



Control flow is the order that instructions are executed in a program.

A control statement is a statement that determines the control flow of a set of instructions.

Types of Control:

- **Sequential control**: Instructions are executed in the order that they are written
- **Selection control**: Selectively executes the instructions. **E.g.** Decision Control
- **Iterative control**: Repeatedly executes the instructions. **E.g.** Loops.

int x = printf ("Hello) → returns the no of characters of the thing that it prints