Functions – Task which is repetitive in nature but needs to be executed at different intervals of time can be isolated / separated & assigned to a function.

Code without Function Code with Function

3B

3A

2B

2A

1B

1A

int main()

{

do this1;

do this2;

fn(); // function call

do this3;

do this4;

do this5;

fn();

do this6;

fn();

do this7

do this8;

}

// function definition

fn()

{  
 do that1;

do that2;

do that3;

do that4;

}

int main()

{

do this1;

do this2;

do that1;

do that2;

do that3;

do that4;

do this3;

do this4;

do this5;

do that1;

do that2;

do that3;

do that4;

do this6;

do that1;

do that2;

do that3;

do that4;

do this7

do this8;

}

When a call is given to a function from main() (reserved function name by the compiler & gets called automatically when the program executes), activities in main() are suspended temporarily & the control of the program is sent to that function. After all statements from that function get executed, the call returns back to main() & executes the code below the function call. Here main() is the “Calling function” & fn() is “Called function”.

Advantages of using functions

1. As the size of code decreases, compilation time also reduces
2. As the code becomes compact, it becomes more easier to maintain
3. It is a good idea to isolate the code with different logic & store it in a function to reduce the complexity of the application
4. Task of writing different functions can be assigned to different developers, these functions can be made a part of different header files & included in the main application. This will definitely bring the application development time OR in other words, increase the productivity.

There are 2 types of functions

1. Library Functions – Like printf(), scanf(), exit(), strcpy(), strlen(), strcmp(), fgetc(), fprintf(), fscanf(), fwrite(), fread(), etc. which come as a part & parcel of std C compiler
2. User Defined Functions – Project specific

There are 2 types of syntax to call a function

1. fn\_name(argument-list);
2. var\_name = fn\_name(argument-list);

argument-list is optional.

The 1st type of functions are called as “void” functions. “void” is a return type of a function. When we say that a function is “void”, it means the function doesn’t return any value to the calling function.