

Q.2 Declare a local variable inside a function and try to access it outside the function. Compare this with accessing the global variable from within the function.

```
# include <stdio.h>
int global Var = 10;
```

```
Void myFunction () {
```

```
    int localVar = 20;
    printf ("Inside function : %n");
    printf ("Local variable = %d/n", localVar);
    printf ("Global variable = %d/n", globalVar);
}
```

```
Int main () {
```

```
    myFunction ();

```

```
    printf ("%n Outside function : %n");

```

```
    printf ("Global variable = %d/n", globalVar);

```

```
    return 0;
}
```

```
}
```

Remarks:

Teacher's Signature _____

The image shows a screenshot of a code editor interface. On the left, the file `main.c` is open, displaying the following C code:

```
1 #include <stdio.h>
2
3 int globalVar = 10;
4
5 void myFunction() {
6     int localVar = 20;
7
8     printf("Inside function:\n");
9     printf("Local variable = %d\n", localVar);
10    printf("Global variable = %d\n", globalVar);
11 }
12
13 int main() {
14     myFunction();
15
16     printf("\nOutside function:\n");
17     // Trying to access 'localVar' here would cause a compilation
18     // error.
19     printf("Global variable = %d\n", globalVar);
20
21 }
22
```

The code includes comments explaining the behavior of local and global variables within functions. The `Run` button is highlighted in blue at the top right. To the right of the code editor is the `Output` pane, which displays the execution results:

```
Inside function:
Local variable = 20
Global variable = 10

Outside function:
Global variable = 10

== Code Execution Successful ==
```