Storage classes

1. Run the following Code and analyse the output.

```
/* storage class and scope */
#include <stdio.h>

void funct1(void);
void funct2(void);

/* external variable, scope is global to main(), funct1() and funct2(), extern
keyword is omitted here, coz just one file */
int globvar = 10;

int main()
{
    printf("\n****storage classes and scope****\n");
    /* external variable */
    globvar = 20;

    printf("\nVariable globvar, in main() = %d\n", globvar);
    funct1();
    printf("\nVariable globvar, in main() = %d\n", globvar);
    funct2();
    printf("\nVariable globvar, in main() = %d\n", globvar);
    return 0;
}

/* external variable, scope is global to funct1() and funct2() */
int globvar2 = 30;

void funct1(void)
{
    /* auto variable, scope local to funct1() and funct1() cannot access the external
globvar */
    char globvar;

    /* local variable to funct1() */
    globvar = 'A';
    /* external variable */
    globvar2 = 40;

    printf("\nIn funct1(), globvar = %c and globvar2 = %d\n", globvar, globvar2);
}

void funct2(void)
{
```

```
    /* auto variable, scope local to funct2(), and funct2() cannot access the
external globvar2 */
    double globvar2;
    /* external variable */
    globvar =  50;
    /* auto local variable to funct2() */
    globvar2 = 1.234;
    printf("\nIn funct2(), globvar = %d and globvar2 = %.4f\n", globvar, globvar2);
}
```

2.What is the output of the following code?

```
#include <stdio.h>
int i;    //By default it is extern variable
int main(){
    printf("%d",i);
    return 0;
}
```

3.A particular extern variable can be declared many times but we can initialize at only one time. For example:

```
    extern int i; //Declaring the variable i.
    int i=25;     //Initializing the variable.
    extern int i; //Again declaring the variable i.
    #include <stdio.h>
    int main(){
    extern int i; //Again declaring the variable i.
    printf("%d",i);
    return 0;
}
```

Check what happens if we initialize *i* it more than once.

4. What is the output of the following code? Do the necessary changes to run the code.

```
#include <stdio.h>
extern int i;
int i=10;   //Initialization statement
i=25;       //Assignment statement
int main(){
```

```
    printf("%d",i);
    return 0;
}
```

5.Compile and execute above two file one.c and two.c at the same time:

```
//one.c

    #include<conio.h>
    int i=25; //By default extern variable
    int j=5;  //By default extern variable
    /**
    Above two line is initialization of variable i and
    j.
    */
    void main(){
            clrscr();
            sum();
            getch();
        }

        //two.c

        #include<stdio.h>
        extern int i; //Declaration of variable i.
        extern int j; //Declaration of variable j.
        /**
        Above two lines will search the initialization
        statement of variable i and j either in two.c
        (if initialized variable is static or extern)
        or one.c (if initialized variable is extern)
        */
        void sum(){
          int s;
        s=i+j;
             printf("%d",s);
        }
```

6. Analyse the output of the following code:

```c
 #include <stdio.h>

static char c;
static int i;
static float f;
static char *str;
int main(){
    printf("%d %d %f %s",c,i,f,str);
    return 0;
}
```