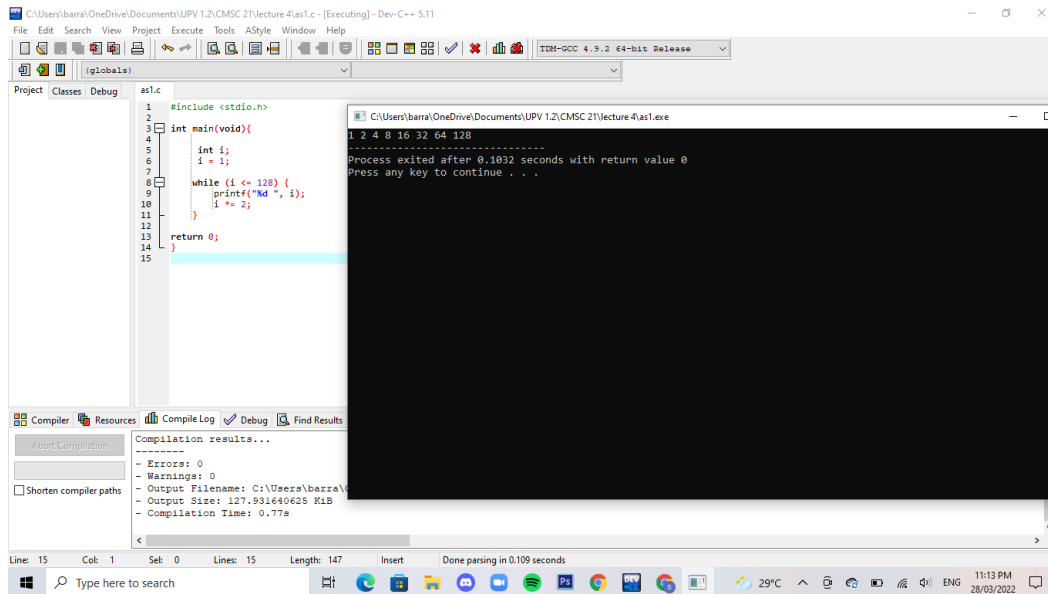


CMSC 21 (Lec) -1

Assignment #4

1.)



```
#include <stdio.h>

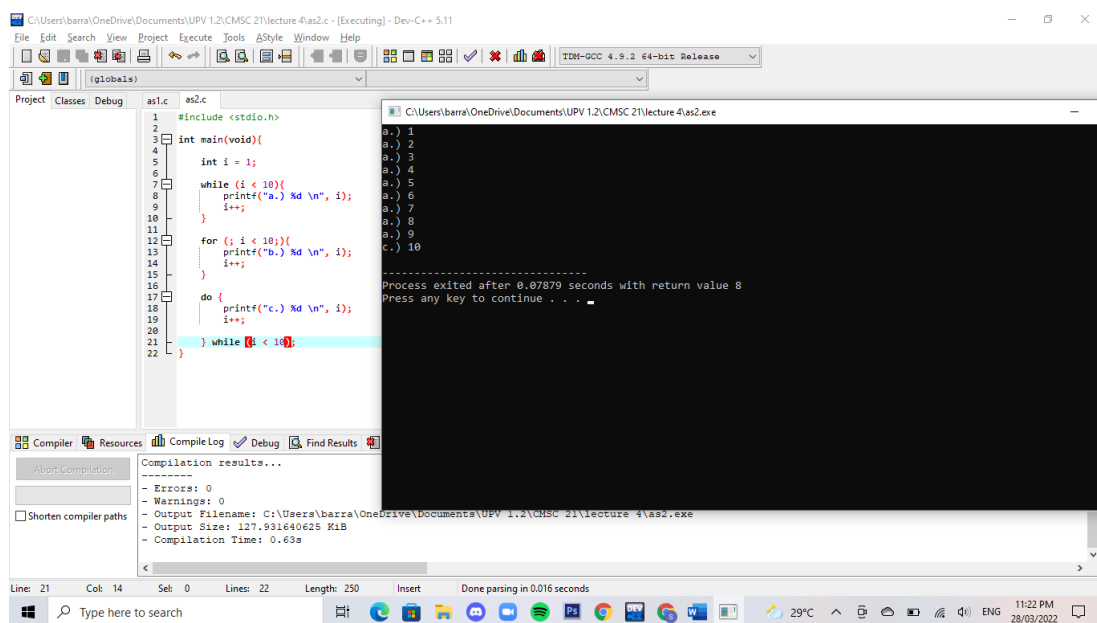
int main(void){
    int i;
    i = 1;
    while (i <= 128){
        printf("%d ", i);
        i *= 2;
    }
    return 0;
}
```

Output: 1 2 4 8 16 32 64 128

Process exited after 0.1032 seconds with return value 0
Press any key to continue . . .

The output of the program is “1 2 4 8 16 32 64 128”

2.)



```
#include <stdio.h>

int main(void){
    int i = 1;
    while (i < 10){
        printf("a.) %d\n", i);
        i++;
    }
    for (; i < 10;){
        printf("b.) %d\n", i);
        i++;
    }
    do {
        printf("c.) %d\n", i);
        i++;
    } while (i < 10);
}
```

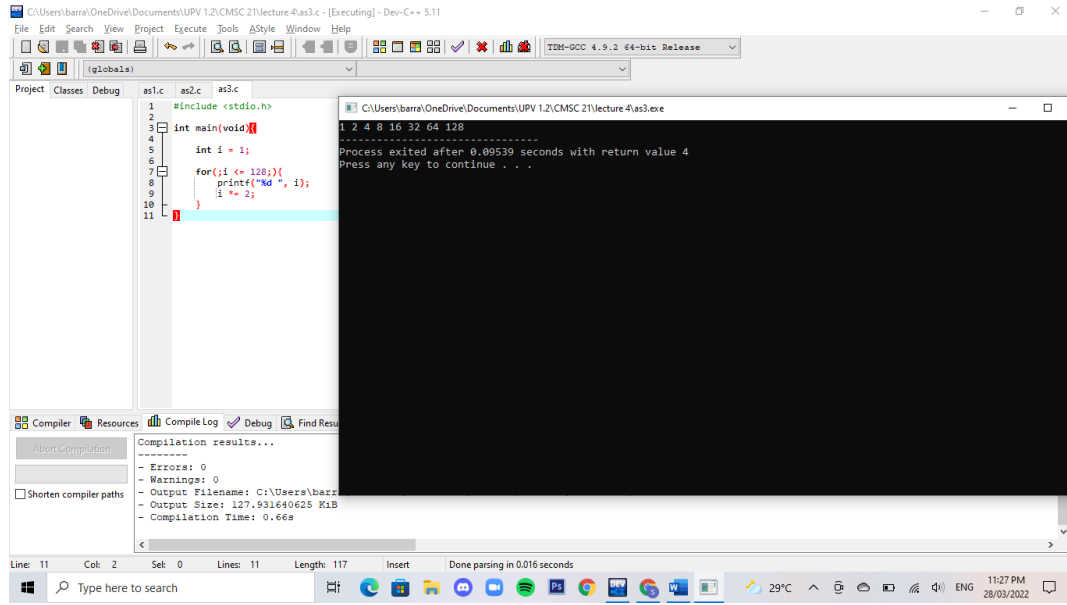
Output: a.) 1
a.) 2
a.) 3
a.) 4
a.) 5
a.) 6
a.) 7
a.) 8
a.) 9
c.) 10

Process exited after 0.07879 seconds with return value 8
Press any key to continue . . .

Letter c (do-while loop) is not equivalent to the other 2 (while & for loops) since it is the only one who printed the number 10.

In the command prompt, you will not see outputs from letter b since letter a already did the task.

3.

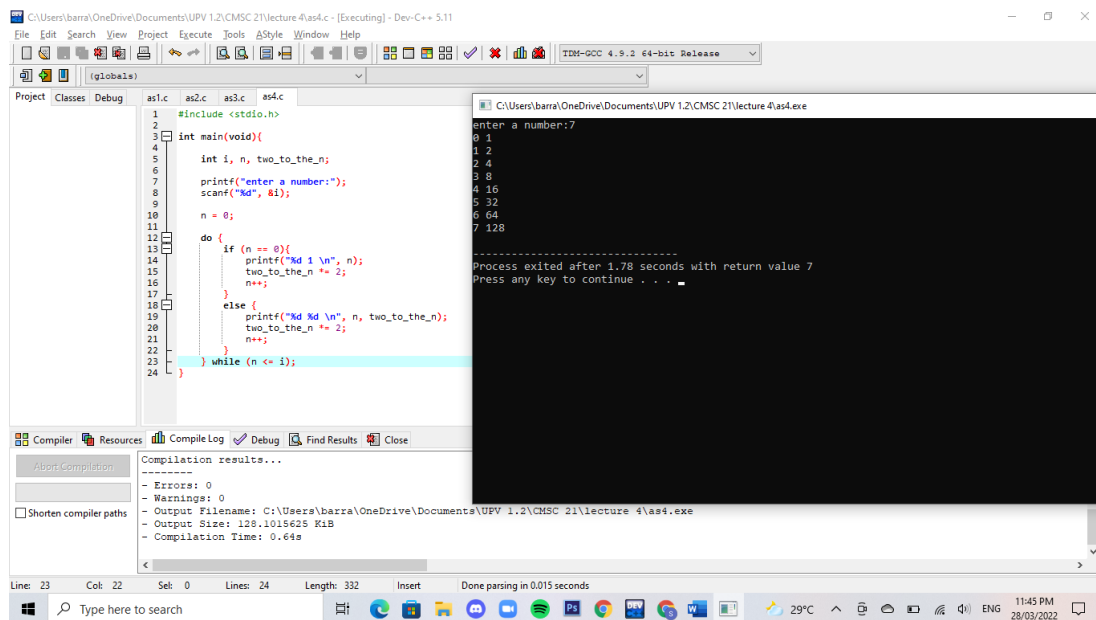


```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     int i = 1;
6     do {
7         printf("%d ", i);
8         i *= 2;
9     } while (i <= 128);
10
11 }
```

Output window:

```
1 2 4 8 16 32 64 128
.....
Process exited after 0.09539 seconds with return value 4
Press any key to continue . . .
```

4.



```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     int i, n, two_to_the_n;
6     printf("enter a number:");
7     scanf("%d", &i);
8     n = 0;
9
10    do {
11        if (n == 0) {
12            printf("%d 1\n", n);
13            two_to_the_n = 2;
14            n++;
15        } else {
16            printf("%d %d\n", n, two_to_the_n);
17            two_to_the_n *= 2;
18            n++;
19        }
20    } while (n <= i);
21
22 }
```

Output window:

```
enter a number:7
0 1
1 2
2 4
3 8
4 16
5 32
6 64
7 128
.....
Process exited after 1.78 seconds with return value 7
Press any key to continue . . .
```

5.

The screenshot shows the Dev-C++ IDE with a C program named `as5.c` open. The program prompts the user to enter the number of days and the starting day of the week, then prints a calendar grid. The execution window shows the user inputting 31 days and 3 as the starting day, resulting in a 6x7 grid of numbers from 1 to 31. The compilation results show no errors or warnings.

```
#include <stdio.h>

int main(void){
    int days, beginning_num, space, row, n;

    printf("Enter number of days:");
    scanf("%d", &days);

    if (days >= 28 && days <= 31){
        printf("Enter the starting day of the week (1=Sun, 7=Sat):");
        scanf("%d", &beginning_num);

        for (row = 1; row < beginning_num; row++){
            printf(" ");
        }

        do{
            space = row % 8;

            if (space == 0){
                printf("\n");
            }
            else{
                printf("%2d ", n);
                n++;
            }
        } while (n <= days);
    }
}
```

Enter number of days:31
Enter the starting day of the week (1=Sun, 7=Sat):3
1 2 3 4 5
6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31

Process exited after 3.134 seconds with return value 31
Press any key to continue . . .

Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\barra\OneDrive\Documents\UPV 1.2\CMSC 21\lecture 4\as5.exe
- Output Size: 129.2734375 KiB
- Compilation Time: 0.67s