

## LECTURE 4

1.

```

c as1.c > main(void)
1  #include <stdio.h>
2
3  int main(void){
4      int i;
5
6      i = 1;
7      while (i <= 128) {
8          printf("%d ", i);
9          i *= 2;
10     }
11
12     return 0;
13
14
12481b52b4128
PS C:\Human-Files\College\UPV_Y1\SEM_2\CMSC21\Lecture4>
1 2 4 8 16 32 64 128

```

Output: 1 2 4 8 16 32 64 128

2.

```

1  #include <stdio.h>
2
3  int main(void){
4      int i;
5
6      i = 0; // test variable
7      printf("VALUE: %d\nWHILE LOOP: ", i);
8      while (i < 10) {
9
10         printf("%d ", i);
11         i++;
12     }
13
14     i = 0; // test variable
15     printf("\nFOR LOOP: ");
16     for (; i < 10;) {
17
18         printf("%d ", i);
19         i++;
20     }
21
22     i = 0; // test variable
23     printf("\nDO WHILE LOOP: ");
24     do{
25
26         printf("%d ", i);
27         i++;
28     } while (i < 10);
29
30     return 0;
31
PS C:\Human-Files\College\UPV_Y1\SEM_2\CMSC21\Lecture4> cc
VALUE: 0
WHILE LOOP: 0 1 2 3 4 5 6 7 8 9
FOR LOOP: 0 1 2 3 4 5 6 7 8 9
DO WHILE LOOP: 0 1 2 3 4 5 6 7 8 9
PS C:\Human-Files\College\UPV_Y1\SEM_2\CMSC21\Lecture4> cc
VALUE: 20
WHILE LOOP:
FOR LOOP:
DO WHILE LOOP: 20

```

The do-while loop is different from the other two because the do-while loop executes the code block before checking the condition unlike the other 2 loops which check the condition first before the code block is executed.

3.

```

C as3.c > main(void)
1  #include <stdio.h>
2
3  int main(void){
4
5      for (int i = 1; i <= 128; i *= 2) {
6          printf("%d ", i);
7      }
8
9      return 0;
10
11 }

```

PS C:\Human-Files\College\21\Lecture4\ ; if (\$?) {  
1 2 4 8 16 32 64 128

4.

```

C as4.c > main(void)
1  #include <stdio.h>
2
3  int main(void) {
4
5      int input, n, two_n = 1;
6
7      while(1){
8
9          printf("\nPOWERS OF TWO\n\nInput n (from 0 - 30 only): ");
10         scanf("%d", &input);
11
12         // exponent of 31 results in miscalculation
13         if (input < 0 || input > 30){
14             printf("Program can only calculate til power of 30.\nPlease try again!");
15         }
16         else{ // input is within 0 - 30
17             break;
18         }
19     }
20
21     printf("  n  2^n\n-----\n");
22     for (n = 0; n <= input; n++) {
23         printf("%3d  %d\n", n, two_n);
24         two_n *= 2;
25     }
26
27     printf("Thanks for using the program! Goodbye");
28
29     return 0;
30
31 }

```

PS C:\Human-Files\College\UPV\_Y1\SEM\_2\CMSC21\Lecture4&gt;

POWERS OF TWO

Input n (from 0 - 30 only): -3  
 Program can only calculate til power of 30.  
 Please try again!

POWERS OF TWO

Input n (from 0 - 30 only): 32  
 Program can only calculate til power of 30.  
 Please try again!

POWERS OF TWO

Input n (from 0 - 30 only): 5

```

  n  2^n
-----
0  1
1  2
2  4
3  8
4  16
5  32

```

Thanks for using the program! Goodbye

```
C as5.c > ...
1  #include <stdio.h>
2
3  /*
4   CALENDAR MAKER
5
6   It is assumed that user will only
7   input integers.
8  */
9
10
11 int main(void){
12
13     int weekday, start_day, days;
14
15     printf("\nCALENDAR MAKER");
16
17     // asks for number of days in the month
18     while (1){
19
20         printf("\nEnter number of days in the month (28-31 only): ");
21         scanf("%d", &days);
22
23         if (days < 28 || days > 31){
24             printf("Invalid input!\nInput can only be from 28 to 31 only."
25                 | " Please Try again.\n");
26         }
27         else {
28             break;
29         }
30     }
31
32     // asks for starting day of the month
33     while (1){
34
35         printf("{1} Sun - {7} Sat\nEnter the starting day of the week: ");
36         scanf("%d", &start_day);
37
38         if (start_day < 1 || start_day > 7){
39             printf("Invalid input!\nInput can only be from 1 to 7 only."
40                 | "Please Try again.\n");
41         }
42         else {
43             break;
44         }
45     }
46 }
```

AGUILAR, CHARLES

```
46
47     printf("Calendar:\nSu Mo Tu We Th Fr Sat\n");
48
49     // so that first day will start on user's chosen day
50     for (weekday = 1; weekday < start_day; weekday++){
51         printf(" ");
52     }
53
54     // variable for end of week/row
55     start_day = ((8 - start_day) % 7);
56
57     for (weekday = 1; weekday <= days; weekday++){
58
59         printf("%2d ", weekday);
60
61         // makes new row to start new week
62         if ((weekday % 7) == start_day){
63
64             printf("\n");
65         }
66     }
67
68     printf("\nThanks for using the program! Goodbye");
69
70     return 0;
71 }
```

```
PS C:\Human-Files\College\UPV_Y1\SEM_2\CMSC21\Lecture4> cd "c
CALENDAR MAKER
Enter number of days in the month (28-31 only): 21
Invalid input!
Input can only be from 28 to 31 only. Please Try again.

Enter number of days in the month (28-31 only): 31
{1} Sun - {7} Sat
Enter the starting day of the week: 2
Calendar:
Su Mo Tu We Th Fr Sat
  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
Thanks for using the program! Goodbye
PS C:\Human-Files\College\UPV_Y1\SEM_2\CMSC21\Lecture4> |
```

**Github Link:** <https://github.com/Techntlinear/CMSC21/tree/main/Lecture4/Assignments>