

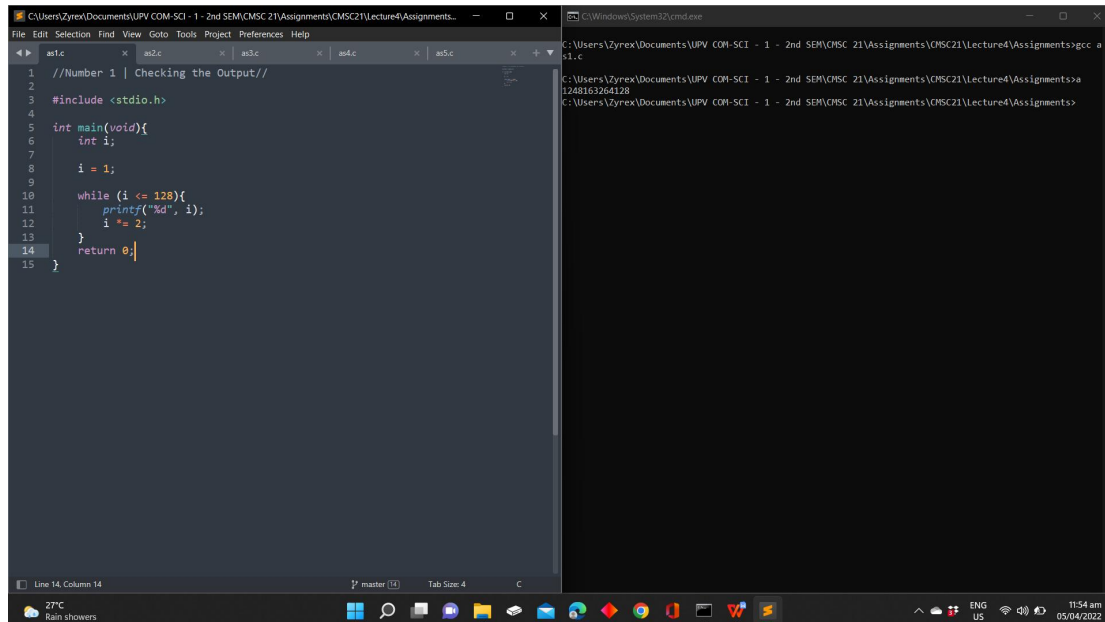
Loop/Repetition Statements

Lecture 4 Assignments

Name: Zyrex Djewel Ganit

1. What is the output of the following program?

The output of this loop is 1248163264128



The screenshot shows a C++ IDE with a file named 'as1.c'. The code is as follows:

```
1 //Number 1 | Checking the Output//
2
3 #include <stdio.h>
4
5 int main(void){
6     int i;
7
8     i = 1;
9
10    while (i <= 128){
11        printf("%d", i);
12        i *= 2;
13    }
14    return 0;
15 }
```

The output window on the right shows the execution result: `C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture4\Assignment>gcc a`
`1248163264128`
`C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture4\Assignment>`

2. Which one of the following statements is not equivalent to the other two (assuming that the loop bodies are the same)?

- a) while (i < 10) {...}
- b) for (; i < 10;) {...}
- c) do {...} while (i < 10);

Among these three types of loops, do while loop is unique as it will execute the block of loops even if it exceeds the required conditions. The other two loops (for and while) will not print or return anything if the input values does not satisfy the condition of the loop. For example, if I enter ten(10), the first two loops will not yield any output and the program would simply stop. However, if I use the body of do while loop, it will first print the value which is ten(10) and then it would check the given condition. Since $10 * 2$ is greater than 10, the loop would now stop resulting with an output of ten(10).

```
21 int main(){
22     int i;
23     printf("Enter a num: ");
24     scanf("%d", &i);
25     for (; i < 10; i+=2){
26         printf("%d", i);
27     }
28 }
29
30 /*int main(){
31     int i;
32     printf("Enter a num: ");
33     scanf("%d", &i);
34     do {
35         printf("%d", i);
36         i+=2;
37     }while(i < 10);
38 }*/
39
40 /* Among these three types of loops, do while loop is unique as it will
41 the required conditions. The other two loops (for and while) will not p
42 not satisfy the condition of the loop. For example, if I enter ten(10),
43 and the program would simply stop. However, if I use the body of do whi
44 is ten(10) and then it would check the given condition. Since 10 * 2 is
45 with an output of ten(10).*/
```

```
21 /*int main(){
22     int i;
23     printf("Enter a num: ");
24     scanf("%d", &i);
25     for (; i < 10; i+=2){
26         printf("%d", i);
27     }
28 }*/
29
30 int main(){
31     int i;
32     printf("Enter a num: ");
33     scanf("%d", &i);
34     do {
35         printf("%d", i);
36         i+=2;
37     }while(i < 10);
38 }
39
40 /* Among these three types of loops, do while loop is unique as it will
41 the required conditions. The other two loops (for and while) will not p
42 not satisfy the condition of the loop. For example, if I enter ten(10),
43 and the program would simply stop. However, if I use the body of do whi
44 is ten(10) and then it would check the given condition. Since 10 * 2 is
45 with an output of ten(10).*/
```

3. Convert item 1 into an equivalent for statement. You can validate your answer by checking if the produced outputs by both the while and for statements are similar.

```
1 //Number 3 | Redefining Num 1 With For Loop//
2
3 #include <stdio.h>
4
5 int main(void){
6     int i;
7     for (i = 1; i <= 128; i++){
8         printf("%d", i);
9         i *= 2;
10     }
11     return 0;
12 }
13
```

4. Write a code that computes for the power of two:
TABLE OF POWERS OF TWO

```

1 //Number 4 | Power of 2//
2
3 #include <stdio.h>
4
5 int main(){
6
7     int num = 2, n, result = 1;
8
9     printf("Enter a number for the exponent: ");
10    scanf("%d", &n);
11
12    while (n){
13        result = result * num;
14        n--;
15    }
16    /*This Loop will compute for the result(Initialized with 1), it will
17    The value would then be stored to result the process will keep on r
18    it will stop if n is equal to 0*/
19    printf("2 to the n is equal to %d", result);
20
21 }

```

```

C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture4\Assignments>gcc a
s4.c
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture4\Assignments>a
s4.c
Enter a number for the exponent: 5
2 to the n is equal to 32

```

5. Write a program that displays a one-month calendar.
There should be a user prompt to set:

- The number of days
- The day of the week on which the month begins.

Additionally, add checkers to validate whether the days entered are valid. For instance, the following number of days are invalid: 32, -1, 0, 27.
This addition will be a good refresher to our previous topic, selection statements.

```

1 //Number 5 | Printing Number With Calendar Format//
2
3 //Importing Functions//
4 #include <stdio.h>
5
6 //Main Function//
7 int main(void){
8
9     //Declaring Variables//
10    int i, num_days, start;
11
12    //Scanning and assigning values for the number of days in a month//
13    printf("\nEnter number of days in month: ");
14    scanf("%d", &num_days);
15
16    //If else statement to check if the number of days in a month is valid or not//
17    //If it is valid if the number of days is greater than 27 and less than or equal to 31//
18    if (num_days <= 31 && num_days > 27){
19        printf("\nNumber of Day is Valid.\n");
20        //Scanning and Assigning Value for the starting day//
21        printf("\nEnter the starting day of the week(1 = Sun, 7 = Sat: ");
22        scanf("%d", &start);
23
24        //Loops for the program//
25        /*This for loop will produce spaces before the numbers depending on the value of the start
26        the condition of this program will compare i minus start and start minus 1 in order for the
27        with the other days below, loop will then increment the value of i*/
28        for (i = 1; i < start - 1; i++){
29            printf(" ");
30        }
31        /*This for loop will output the remaining number of days*/
32        for (i = 1; i <= num_days; i++){
33            //
34            printf("%3d ", i); //There will 2 spaces before the numbers to separate them.
35            if ((i + start - 1) % 7 == 0){
36                /*If this condition have 0 remainder,
37                it will simply proceed to the nextline for the following processes*/
38                printf("\n");
39            }
40        }
41    }
42
43    //If the value of days in the month exceeds the conditions//
44    else{
45        printf("\nInvalid Number of Days.");
46    }
47
48    return 0;
49 }

```

```

C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture4\Assignments>a
s5.c
Enter number of days in month: 30
Number of Day is Valid.
Enter the starting day of the week(1 = Sun, 7 = Sat: 2
 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
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture4\Assignments>a
s5.c
Enter number of days in month: 3
Invalid Number of Days.
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture4\Assignments>

```