

LECTURE 4 ASSIGNMENT LOOPS

1. The output is 1 2 4 8 16 32 64 128 .

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

2. In this code, when you run them independently, they will generate the same output which is 1 2 4 8 16 .

```
2  #include <stdio.h>
3  #include <stdlib.h>
4  // where the program begins
5  int main(void)
6  {
7      // declaration of variable
8      int i = 1;
9      // the while loop
10     while (i < 10){
11         printf ("%d ", i);
12         i *= 2;
13     }
14     // the for loop
15     for (i; i < 10; i *= 2){
16         printf ("%d ", i);
17     }
18     // the do-while loop
19     do {
20         printf ("%d ", i);
21         i *= 2;
22     }while (i < 10);
23
24     return 0;
```

3.

```

1 // the equivalent for statement in item 1
2 // Preprocessor Directives
3 #include <stdio.h>
4 #include <stdlib.h>
5
6 int main(void)
7 {
8     int i = 1;
9     // i is the initialize statement, i<=128 is the condition and i*=2 is where code will be updated
10    // just revising the form to make it a for loop, but the output will still be the same
11    for (i; i<= 128; i*=2){
12        printf ("%d ", i); // the same print statement
13    }
14    return 0;
15 }

```

4.

```

1 // Preprocessor Directives
2 #include <stdio.h>
3 #include <stdlib.h>
4 // where the program begins
5 int main(void)
6 {
7     // declaring of variables using int as the variable type
8     int exponent, value;
9     printf ("Enter exponent value: "); // printing of instruction
10    scanf ("%d", &exponent); // storing the value in the variable exponent
11    // for loop
12    // value=1 as the initialize statement, exponent !=0 is the condition, and exponent-- is the condition where
13    for (value = 1; exponent != 0; exponent--){ // as long as the value of the exponent is not 0, the code will j
14        // decrementing the value of the exponent as the code is updating so that every allue will be coomputed
15        value = value * 2; // the formula in computing the powers of two
16    }
17    // printing the result
18    printf ("The result is: %d", value);
19    return 0;
20 }

```

5.

```

1 // Preprocessor Directives
2 #include <stdio.h>
3 #include <stdlib.h>
4 // where the program begins
5 int main(void)
6 {
7     // declaring of variables using int variable type
8     int days, start, i;
9
10    // printing the instruction
11    printf ("Enter the starting day of the Week (1- Sun and 7- Sat): ");
12    scanf ("%d", &start); // storing the value in the variable start
13    printf ("Enter number of days in a Month: "); // printing the instruction on how many days the user wants
14    scanf ("%d", &days); // storing the value in the variable days
15    printf ("\n");
16    // a conditional on where the program will generate an invalid input if it satisfies the condition
17    // didn't include 29 since February can have 29 days in a month
18    if (days < 1 || days < 28 || days > 31){
19        printf ("Invalid number of days input.\n");
20    }
21    // else statement
22    else{
23        printf ("Here is your calendar:\n");
24        // this code is for the position of the first date (1); where will be dependent on the value of variab
25        // for loop in where; i=1 is the initialize statement, i<start is the condition and i++ is where the loop
26        // this code will run as long as the condition is true
27        for (i = 1; i < start; i++){
28            printf ("    "); // putting four spaces in each number
29        }
30        // this section is for all the dates for one month; where will be dependent to the variable days
31        // for loop in where; i=1 is the initialize statement, i<=days is the condition and i++ is where the loop
32        // this code will run as long as the condition is true
33        for (i = 1; i <= days; i++){
34            printf ("%4d", i); // putting fours spaces in each number to match the spaces of the first date
35            // not setting them the same will generate a chaos in the output
36            if ((start + i - 1) % 7 == 0) // condition in which will evaluate the number, since we have 7 day
37                printf ("\n"); // if the number when evaluated will have no remainder the next number will pro
38        } // for example (3 + 12 - 1) = 14, so when divided by 7 it will generate 0 remainder thus number 1
39    }
40    return 0;
41 }

```