Glen Andrew C. Bulaong

2021-00252

BS Computer Science 1

CMSC 21

Section 1

1.

```c
#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

2.

```c
#include <stdio.h>

int main (void){
    const char NEWLINE = '\n';
    int i;

    //A
    i = 11;
    while (i < 10){
        printf("%d ", i);
        i++;
    }
    printf("%c", NEWLINE);

    //B
    for (i = 11; i < 10; i++){
        printf("%d ", i);
    }
    printf("%c", NEWLINE);

    //C
    i = 11 ;
    do {
        printf("%d ", i);
        i++;
    }
    while(i < 10);

    return 0;
}
```

- If we are going to have the example above, C is not equivalent to the other two. It is because it will execute the lines below "do" at least once unlike its counterparts. Thus, C yields an output when the code is performed.

3.

```c
#include <stdio.h>

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

    return 0;
}
```

Output: 1 2 4 8 16 32 64 128

4.

```c
#include <stdio.h>
#include <math.h>

int main (void) {
    const char NEWLINE = '\n';
    const char TAB = '\t';
    int new_num;
    printf("TABLE OF POWERS OF TWO %c%c", NEWLINE, NEWLINE);
    printf("n  %c  2 to the n %c", TAB, NEWLINE);
    printf("--- %c  ---------- %c", TAB, NEWLINE);

    for (int i = 0; i < 11; i++){
        new_num = pow(2, i);                                    //pow() computes the exponent value for each value of i
        printf("%d %c  %d %c", i, TAB, new_num, NEWLINE);
    }

    return 0;
}
```

Output:

```
C:\Users\glena\Desktop>as4
TABLE OF POWERS OF TWO

n          2 to the n
---        ----------
0          1
1          2
2          4
3          8
4          16
5          32
6          64
7          128
8          256
9          512
10         1024
```

5.

```c
#include <stdio.h>

int main(void){
    const char NEWLINE = '\n';
    const char TAB = '\t';
    const char EMPTY = ' ';
    int days, start;
    int space = 1;                                          //Provide spaces when day is not used.
    int day = 1;                                            //Starting day

    printf("Enter number of days in month: ");
    scanf("%d", &days);                                     //Prompt user for number of days.
    printf("Enter the starting day of the week (1=Sun, 7=Sat): ");
    scanf("%d", &start);                                    //Prompt user what day is the starting day.

    if((days < 28 || days > 31) || (start < 1 || start > 7)){
        /*Judges if user inputted an illegal value.
        If the value is illegal, it will display a warning,
        Otherwise, the program will yield the appropriate calendar. */
        printf("Invalid input");
    }
    else{
        while (day <= days){
            /*The loop will continue until day is less than the desired number of days. */
            for (int i = 0; i < 7; i++){
                /* This will print the days inside the week */
                if (space < start){
                    /* Check if the day in the first week is the starting day.
                    If it is not, print an empty space,
                    Otherwise print the corresponding date. */
                    printf("%c%c", EMPTY, TAB);
                    space++;
                }
                else{
                    if (day > days){
                        /*Check if the value of day overflows the number of days.
                        If the day is over, it will print an empty space. */
                        printf("%c%c", EMPTY, TAB);
                    }
                    else{
                        printf("%d%c", day, TAB);
                    }
                    day++;
                }
            }
            printf("%c", NEWLINE);
        }
    }

    return 0;
}
```

Output:

```
Enter number of days in month: 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
```

Github Link:
https://github.com/dreeew05/CMSC21/tree/master/Lecture4/Assignments