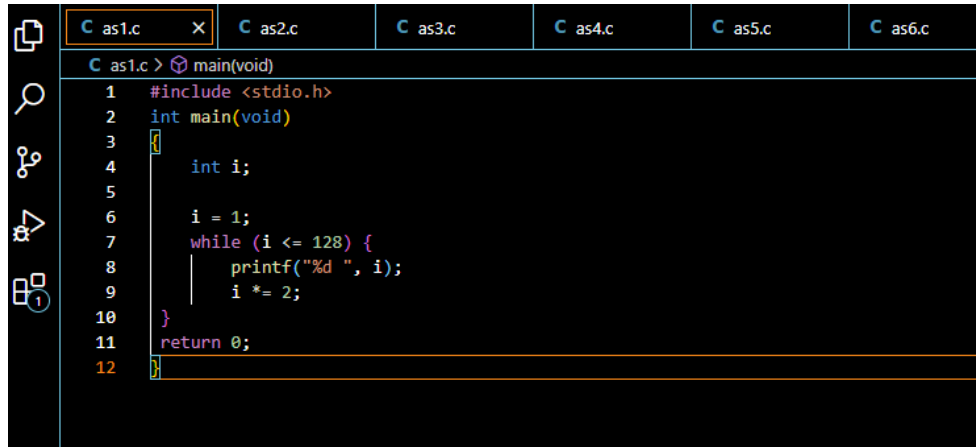


Loops and Arrays

Lecture 4 Assignments

1. What is the output of the following program?



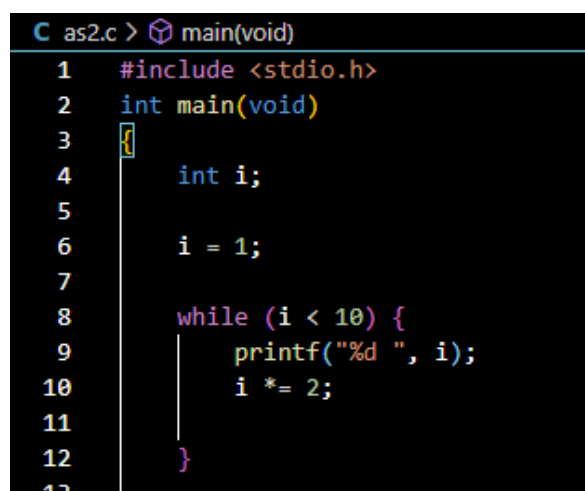
```
C as1.c > main(void)
1  #include <stdio.h>
2  int main(void)
3  {
4      int i;
5
6      i = 1;
7      while (i <= 128) {
8          printf("%d ", i);
9          i *= 2;
10     }
11     return 0;
12 }
```

Output:

1 2 4 8 16 32 64 128

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) { ... }



```
C as2.c > main(void)
1  #include <stdio.h>
2  int main(void)
3  {
4      int i;
5
6      i = 1;
7
8      while (i < 10) {
9          printf("%d ", i);
10         i *= 2;
11     }
12 }
```

b) for (; i < 10;) {...}

```
13  */
14  |   for (; i < 10;) {
15  |       printf("%d ", i);
16  |       i *= 2;
17  |   }
18
19  */
```

c) do {...} while (i < 10);

```
18
19  */ do {
20  |     printf("%d ", i);
21  |     i *= 2;
22  | }
23  | while (i < 10);
24
25
```

Output for a, b, c:

```
1 2 4 8
```

-Since all of the statements of a, b, & c result to the same output, the statements are equivalent of one another

3. Convert item 1 into an equivalent for statement.

```
C as1.c  C as2.c  C as3.c X  C as4.c
C as3.c > main(void)
1  #include <stdio.h>
2  int main(void)
3  {
4      int i;
5
6      for ( i = 1; i <= 128; i *= 2) {
7          printf("%d ", i);
8      }
9      return 0;
10 }
```

Output:

```
1 2 4 8 16 32 64 128
```

4. Write a code that computes for the power of two:

```
C as1.c C as2.c C as3.c C as4.c 1 X C a
C as4.c > main()
1 int main() {
2     long int num;
3     int n;
4     printf("TABLE OF POWERS OF TWO");
5     printf("\nn 2 to the n");
6     printf("\n-- -----");
7     num = 1;
8
9     for (n = 0; n < 11; ++n) {
10        if (n == 0)
11            num = 1;
12        else
13            num = num * 2;
14        printf("\n%d %d", n, num);
15    }
16    return 0;
17 }
```

```
osy.ya4 --dbgExe=C:\MINGW\mingw64\l
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. Write a program that displays a one-month calendar..

```
C as1.c C as2.c C as3.c C as4.c 1 X C a
C as4.c > main()
1 #include<stdio.h>
2 int main(){
3     int i,days;
4     int first;
5
6     printf("Enter number of days in month:\n");
7     scanf("%d",&days);
8     printf("Enter the starting day of the week (1=Sun, 7=Sat):\n");
9     scanf("%d",&first);
10
11     for(i=1;i<first;i++)
12         printf(" ");
13
14     if(days>31 || days<=0 || days<28){
15         printf("Error! The number of days you have entered is invalid. Please put a number from 27-31");
16     }
17     else{
18         for(i=1;i<=days;i++){
19             printf("%3d",i);
20             if((first+i-1)%7==0)
21                 printf("\n");
22         }
23     }
24     return 0;
25
26 }
```

Checking:

Enter number of days in month:
31
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	31			

Enter number of days in month:
32
Enter the starting day of the week (1=Sun, 7=Sat):
1
Error! The number of days you have entered is invalid. Please put a number from 27-31

Enter number of days in month:
-1
Enter the starting day of the week (1=Sun, 7=Sat):
3
Error! The number of days you have entered is invalid. Please put a number from 27-31

Enter number of days in month:
27
Enter the starting day of the week (1=Sun, 7=Sat):
6
Error! The number of days you have entered is invalid. Please put a number from 27-31

6.

a. Revise line 16 such that you use a designated initializer to set pathways 0 and 2 to true, and the rest will be false. Make the initializer as short as possible.

```
C as6.c > main()
1  #include <stdio.h>
2  #include <stdbool.h>
3
4  #define NUM_PATHWAYS ((int) (sizeof(pathway) / sizeof(pathway[0])))
5
6  int main(){
7
8      bool pathway[8]= {[0]= 1, [2]= 1};
9      //bool pathway[8]= {1, 0, 1}; //
10
11     for (int i = 0; i < NUM_PATHWAYS; i++){
12         if (pathway[i]){
13             printf("pathway[%d] is open \n", i);
14         }else{
15             printf("pathway[%d] is close \n", i);
16         }
17     }
18
19     return 0;
```

```
3vd.42f' '--dbgExe=C:\MinGW\mingw64
pathway[0] is open
pathway[1] is close
pathway[2] is open
pathway[3] is close
pathway[4] is close
pathway[5] is close
pathway[6] is close
pathway[7] is close
```

b. Revise line 16 such that the initializer will be short as possible (without using a designated initializer).

```
C as6.c > main()
1  #include <stdio.h>
2  #include <stdbool.h>
3
4  #define NUM_PATHWAYS ((int) (sizeof(pathway) / sizeof(pathway[0])))
5
6  int main(){
7
8      //bool pathway[8]= {[0]= 1, [2]= 1}; //
9      bool pathway[8]= {1, 0, 1};
10
11     for (int i = 0; i < NUM_PATHWAYS; i++){
12         if (pathway[i]){
13             printf("pathway[%d] is open \n", i);
14         }else{
15             printf("pathway[%d] is close \n", i);
16         }
17     }
18
19     return 0;
```

7. A road network can be represented using graphs. Assuming we have points / stations a, b, c, d, e, f, g, and h, we can represent a direct path from a point to another point using arrows.

```

Adjacency Matrix
  A  B  [C] [D]  E  F  G  H  I
A  1  1  0  0  0  1  0  0  0
B  1  1  1  0  0  0  0  0  0
[C] 0  1  1  0  1  1  0  0  1
[D] 0  0  0  1  1  0  0  0  0
E  0  0  0  1  1  0  0  0  0
F  1  0  1  0  0  1  0  0  0
G  1  0  0  1  0  0  1  0  0
H  0  0  0  0  0  0  0  1  1
I  0  0  0  0  0  0  0  1  1

At which point are you currently located (0 - A, 1 - B, 2 - C, 3 - D, 4 - E, 5 - F, 6 - G, 7 - H, 8 - I): 1

You are currently at Point B.
The nearest charging station to Point B is Point C.
  
```

```

Adjacency Matrix
  A  B  [C] [D]  E  F  G  H  I
A  1  1  0  0  0  1  0  0  0
B  1  1  1  0  0  0  0  0  0
[C] 0  1  1  0  1  1  0  0  1
[D] 0  0  0  1  1  0  0  0  0
E  0  0  0  1  1  0  0  0  0
F  1  0  1  0  0  1  0  0  0
G  1  0  0  1  0  0  1  0  0
H  0  0  0  0  0  0  0  1  1
I  0  0  0  0  0  0  0  1  1

At which point are you currently located (0 - A, 1 - B, 2 - C, 3 - D, 4 - E, 5 - F, 6 - G, 7 - H, 8 - I): 5

You are currently at Point F.
The nearest charging station to Point F is Point C.
  
```

\