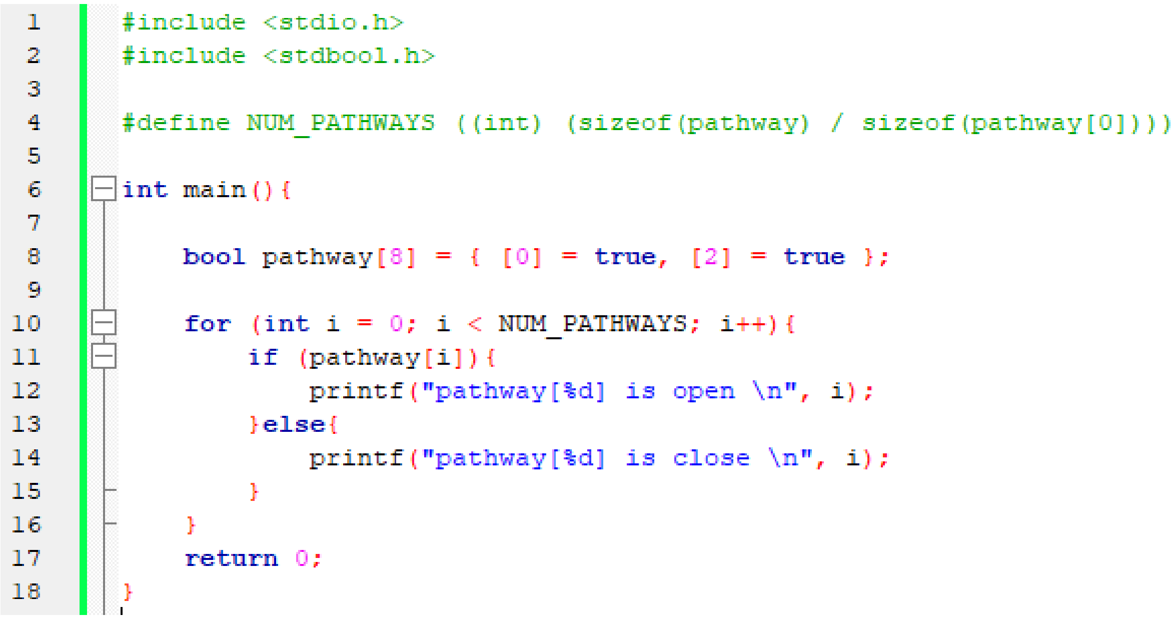
Precious Anne D. Ramil

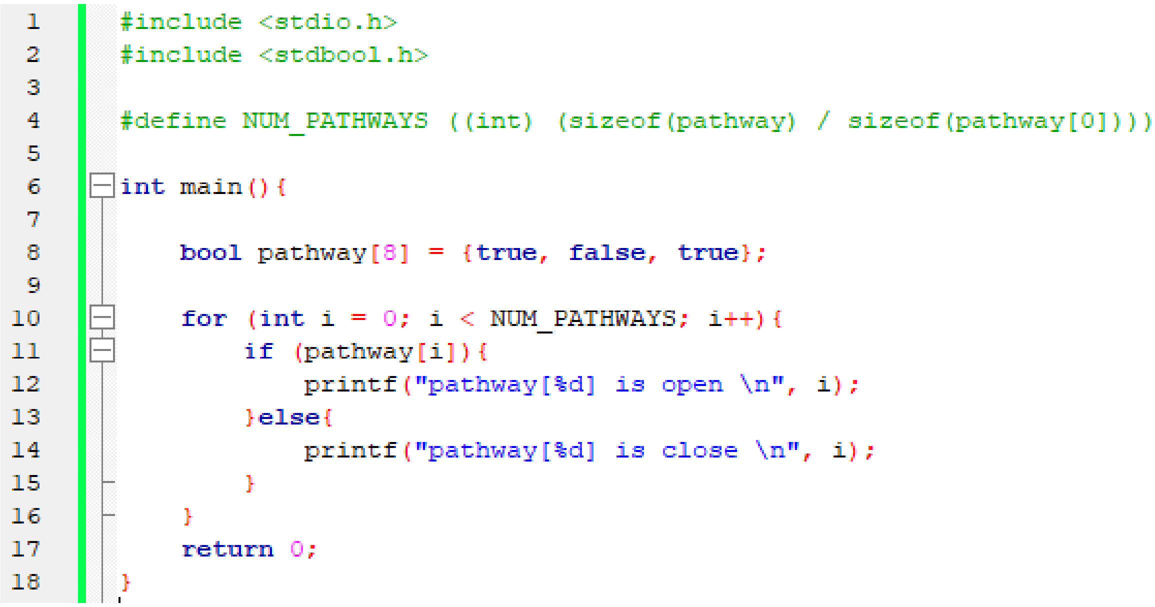
CMSC21-2

1. a) **Answer:** bool pathway[8] = { [0] = true, [2] = true }



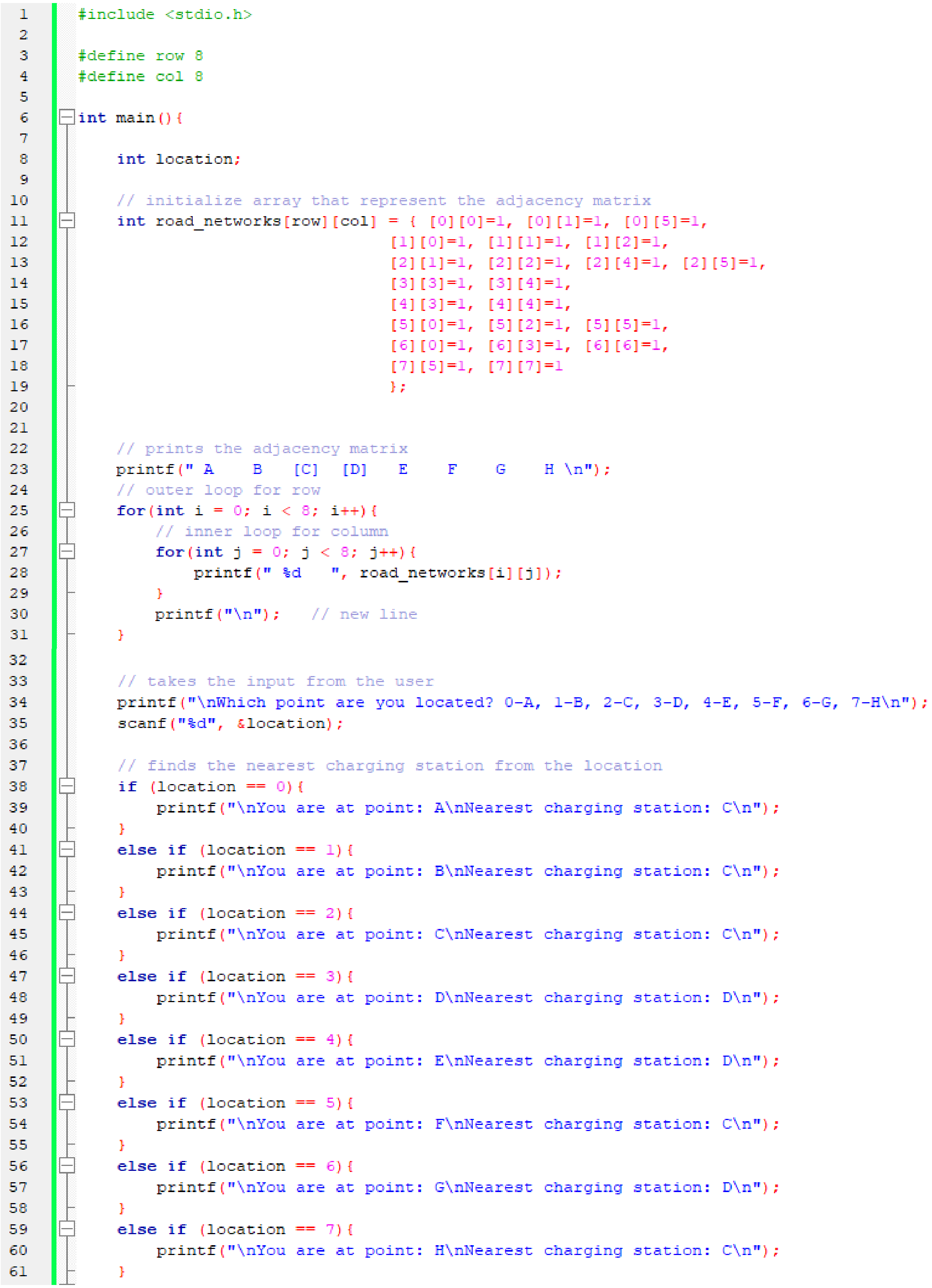
```
#include <stdio.h>
#include <stdbool.h>

#define NUM_PATHWAYS ((int) (sizeof(pathway) / sizeof(pathway[0])))

int main(){

    bool pathway[8] = { [0] = true, [2] = true };

    for (int i = 0; i < NUM_PATHWAYS; i++){
        if (pathway[i]){
            printf("pathway[%d] is open \n", i);
        }else{
            printf("pathway[%d] is close \n", i);
        }
    }
    return 0;
}
```

b) **Answer:** bool pathway[8] = {true, false, true}

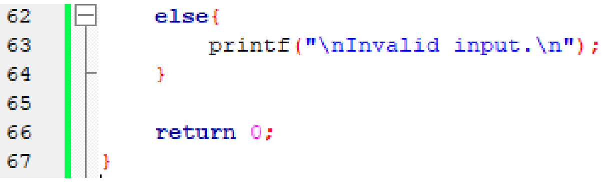


```
#include <stdio.h>
#include <stdbool.h>

#define NUM_PATHWAYS ((int) (sizeof(pathway) / sizeof(pathway[0])))

int main(){

    bool pathway[8] = {true, false, true};

    for (int i = 0; i < NUM_PATHWAYS; i++){
        if (pathway[i]){
            printf("pathway[%d] is open \n", i);
        }else{
            printf("pathway[%d] is close \n", i);
        }
    }
    return 0;
}
```

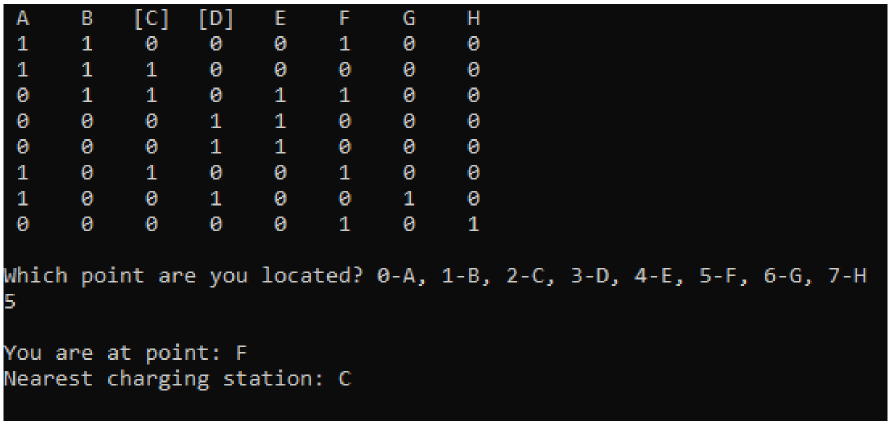
2.



```c
#include <stdio.h>

#define row 8
#define col 8

int main(){

    int location;

    // initialize array that represent the adjacency matrix
    int road_networks[row][col] = { [0][0]=1, [0][1]=1, [0][5]=1,
                                [1][0]=1, [1][1]=1, [1][2]=1,
                                [2][1]=1, [2][2]=1, [2][4]=1, [2][5]=1,
                                [3][3]=1, [3][4]=1,
                                [4][3]=1, [4][4]=1,
                                [5][0]=1, [5][2]=1, [5][5]=1,
                                [6][0]=1, [6][3]=1, [6][6]=1,
                                [7][5]=1, [7][7]=1
                                };


    // prints the adjacency matrix
    printf(" A    B   [C]  [D]   E    F    G    H \n");
    // outer loop for row
    for(int i = 0; i < 8; i++){
        // inner loop for column
        for(int j = 0; j < 8; j++){
            printf(" %d   ", road_networks[i][j]);
        }
        printf("\n");   // new line
    }

    // takes the input from the user
    printf("\nWhich point are you located? 0-A, 1-B, 2-C, 3-D, 4-E, 5-F, 6-G, 7-H\n");
    scanf("%d", &location);

    // finds the nearest charging station from the location
    if (location == 0){
        printf("\nYou are at point: A\nNearest charging station: C\n");
    }
    else if (location == 1){
        printf("\nYou are at point: B\nNearest charging station: C\n");
    }
    else if (location == 2){
        printf("\nYou are at point: C\nNearest charging station: C\n");
    }
    else if (location == 3){
        printf("\nYou are at point: D\nNearest charging station: D\n");
    }
    else if (location == 4){
        printf("\nYou are at point: E\nNearest charging station: D\n");
    }
    else if (location == 5){
        printf("\nYou are at point: F\nNearest charging station: C\n");
    }
    else if (location == 6){
        printf("\nYou are at point: G\nNearest charging station: D\n");
    }
    else if (location == 7){
        printf("\nYou are at point: H\nNearest charging station: C\n");
    }
```



```c
    else{
        printf("\nInvalid input.\n");
    }

    return 0;
}
```



```
 A    B   [C]  [D]   E    F    G    H
 1    1    0    0    0    1    0    0
 1    1    1    0    0    0    0    0
 0    1    1    0    1    1    0    0
 0    0    0    1    1    0    0    0
 0    0    0    1    1    0    0    0
 1    0    1    0    0    1    0    0
 1    0    0    1    0    0    1    0
 0    0    0    0    0    1    0    1

Which point are you located? 0-A, 1-B, 2-C, 3-D, 4-E, 5-F, 6-G, 7-H
5

You are at point: F
Nearest charging station: C
```

Github link: https://github.com/pdramil/CMSC21/tree/main/Lecture%206-7/Assignments