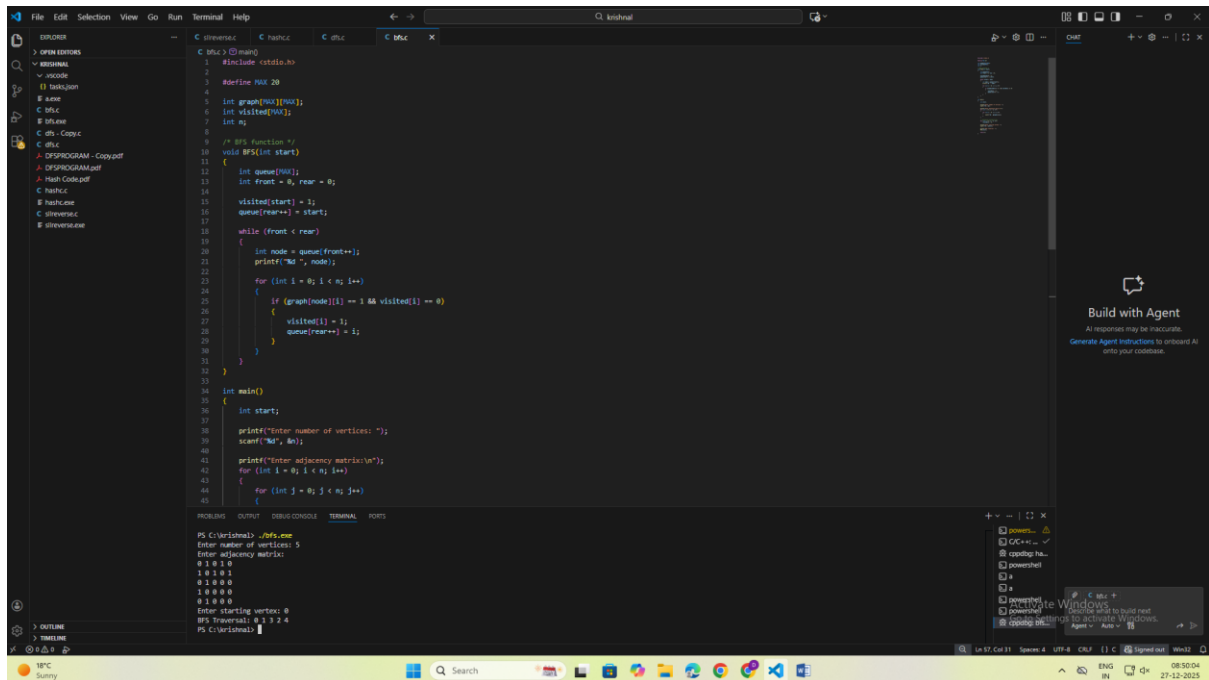
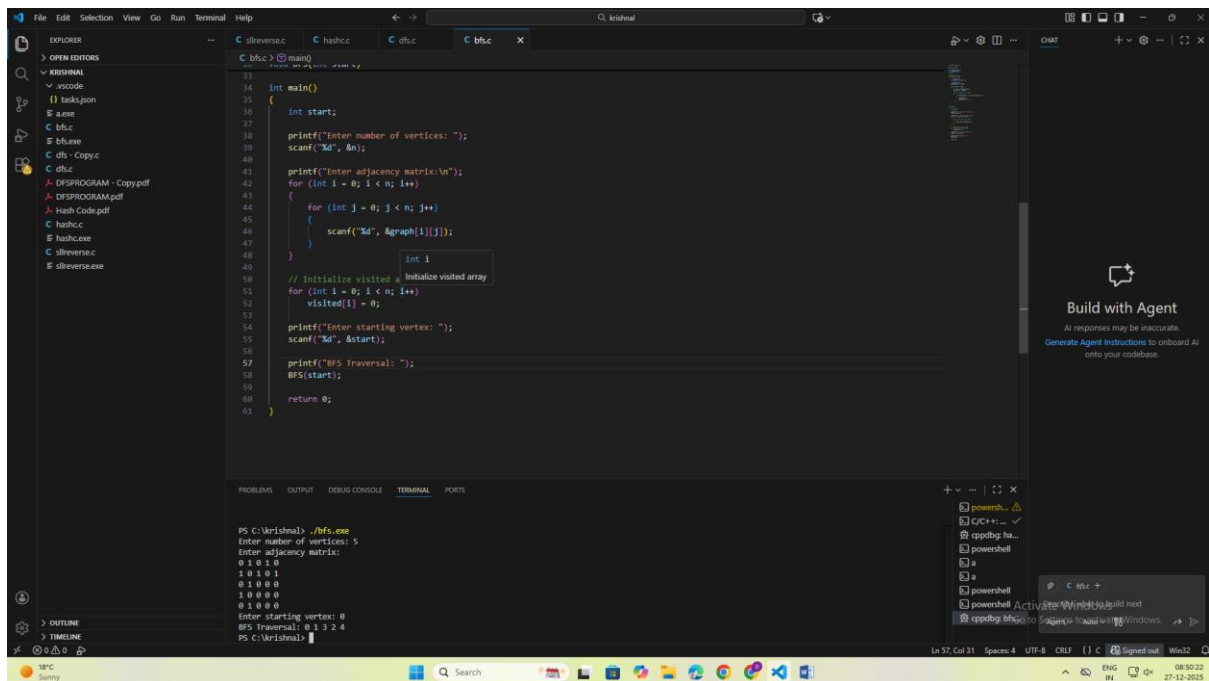


## 9(a) BFS program



```
1 #include <iostream>
2
3 #define MAX 20
4
5 int graph[MAX][MAX];
6 int visited[MAX];
7 int n;
8
9 // BFS function
10 void BFS(int start)
11 {
12     int queue[MAX];
13     int front = 0, rear = 0;
14
15     visited[start] = 1;
16     queue[rear++] = start;
17
18     while (front < rear)
19     {
20         int node = queue[front++];
21         printf("%d ", node);
22
23         for (int i = 0; i < n; i++)
24         {
25             if (graph[node][i] == 1 && visited[i] == 0)
26             {
27                 visited[i] = 1;
28                 queue[rear++] = i;
29             }
30         }
31     }
32 }
33
34 int main()
35 {
36     int start;
37
38     printf("Enter number of vertices: ");
39     scanf("%d", &n);
40
41     printf("Enter adjacency matrix:\n");
42     for (int i = 0; i < n; i++)
43     {
44         for (int j = 0; j < n; j++)
45         {
46             graph[i][j] = 0;
47         }
48     }
49
50     printf("Enter starting vertex: ");
51     scanf("%d", &start);
52
53     BFS(start);
54
55     return 0;
56 }
```

PS C:\Verishal> .\bfs.exe  
Enter number of vertices: 5  
Enter adjacency matrix:  
0 1 0 0  
1 0 1 0  
0 1 0 0  
0 0 0 0  
0 1 0 0  
Enter starting vertex: 0  
BFS Traversal: 0 1 3 2 4  
PS C:\Verishal>



```
1 // BFS function
2 void BFS(int start)
3 {
4     int queue[MAX];
5     int front = 0, rear = 0;
6
7     visited[start] = 1;
8     queue[rear++] = start;
9
10    while (front < rear)
11    {
12        int node = queue[front++];
13        printf("%d ", node);
14
15        for (int i = 0; i < n; i++)
16        {
17            if (graph[node][i] == 1 && visited[i] == 0)
18            {
19                visited[i] = 1;
20                queue[rear++] = i;
21            }
22        }
23    }
24 }
25
26 int main()
27 {
28     int start;
29
30     printf("Enter number of vertices: ");
31     scanf("%d", &n);
32
33     printf("Enter adjacency matrix:\n");
34     for (int i = 0; i < n; i++)
35     {
36         for (int j = 0; j < n; j++)
37         {
38             graph[i][j] = 0;
39         }
40     }
41
42     printf("Enter starting vertex: ");
43     scanf("%d", &start);
44
45     BFS(start);
46
47     return 0;
48 }
```

PS C:\Verishal> .\bfs.exe  
Enter number of vertices: 5  
Enter adjacency matrix:  
0 1 0 0  
1 0 1 0  
0 1 0 0  
0 0 0 0  
0 1 0 0  
Enter starting vertex: 0  
BFS Traversal: 0 1 3 2 4  
PS C:\Verishal>