

PROBLEM 1 : Wave Form Traversal

```
Q1.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int N, M;
6      cin >> N >> M;
7
8      int arr[N][M];
9      for(int i = 0; i < N; i++) {
10         for(int j = 0; j < M; j++) {
11             cin >> arr[i][j];
12         }
13     }
14     for(int j = 0; j < M; j++) {
15         if(j % 2 == 0) {
16             for(int i = 0; i < N; i++) {
17                 cout << arr[i][j] << " ";
18             }
19         } else {
20             for(int i = N-1; i >= 0; i--) {
21                 cout << arr[i][j] << " ";
22             }
23         }
24     }
25 }
26
27 return 0;
28 }
```

```
3
3
1
2
3
4
5
6
7
8
9
1 4 7
8 5 2
3 6 9
```

PROBLEM 2: Transpose Of a Matrix

```
Q2.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int arr[3][3];
6
7      cout << "Enter elements of a 3x3 matrix:\n";
8      for (int i = 0; i < 3; i++) {
9          for (int j = 0; j < 3; j++) {
10             cin >> arr[i][j];
11         }
12     }
13
14     cout << "Your matrix is:\n";
15     for (int i = 0; i < 3; i++) {
16         for (int j = 0; j < 3; j++) {
17             cout << arr[i][j] << " ";
18         }
19         cout << endl;
20     }
21
22     cout << "Transpose of your matrix is:\n";
23     for (int i = 0; i < 3; i++) {
24         for (int j = 0; j < 3; j++) {
25             cout << arr[j][i] << " ";
26         }
27         cout << endl;
28     }
29
30     return 0;
31 }
```

Enter elements of a 3x3 matrix:

1
2
3
4
5
6
7
8
9

Your matrix is:

1 2 3
4 5 6
7 8 9

Transpose of your matrix is:

1 4 7
2 5 8
3 6 9

PROBLEM 3: Spiral Traversal of a Matrix

```
Q3.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int N, M;
6      cin >>N>>M;
7
8      int arr[100][100];
9      for(int i = 0; i < N; i++)
10         for(int j = 0; j < M; j++)
11             cin >> arr[i][j];
12
13     int top = 0, bottom = N - 1;
14     int left = 0, right = M - 1;
15
16     while(top <= bottom && left <= right) {
17
18         for(int j = left; j <= right; j++)
19             cout << arr[top][j] << " ";
20         top++;
21
22         for(int i = top; i <= bottom; i++)
23             cout << arr[i][right] << " ";
24         right--;
25
26         if(top <= bottom) {
27             for(int j = right; j >= left; j--)
28                 cout << arr[bottom][j] << " ";
29             bottom--;
30         }
31         if(left <= right) {
32             for(int i = bottom; i >= top; i--)
33                 cout << arr[i][left] << " ";
34             left++;
35         }
36     }
37     return 0;
38 }
```

```
3
3
1
2
3
4
5
6
7
8
9
1 2 3 6 9 8 7 4 5
```

PROBLEM 4: Rotate Matrix by 90deg clockwise

```
Q4.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int N;
6      cin >> N;
7
8      int arr[100][100];
9
10     for(int i = 0; i < N; i++)
11         for(int j = 0; j < N; j++)
12             cin >> arr[i][j];
13
14     for(int i = 0; i < N; i++) {
15         for(int j = i + 1; j < N; j++) {
16             swap(arr[i][j], arr[j][i]);
17         }
18     }
19     for(int i = 0; i < N; i++) {
20         int start = 0, end = N - 1;
21         while(start < end) {
22             swap(arr[i][start], arr[i][end]);
23             start++;
24             end--;
25         }
26     }
27     for(int i = 0; i < N; i++) {
28         for(int j = 0; j < N; j++)
29             cout << arr[i][j] << " ";
30         cout << endl;
31     }
32
33     return 0;
34 }
```

```
3
1
2
3
4
5
6
7
8
9
7 4 1
8 5 2
9 6 3
```