

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface running on a Mac. The main area displays a C file named `main.c` with the following code:

```
#include <stdio.h>
int main() {
    int rows, cols, i, j;
    printf("Enter number of rows and columns: ");
    scanf("%d %d", &rows, &cols);

    int matrix[rows][cols];
    printf("Enter elements of the matrix:\n");
    for(i = 0; i < rows; i++) {
        for(j = 0; j < cols; j++) {
            scanf("%d", &matrix[i][j]);
        }
    }

    printf("Matrix is:\n");
    for(i = 0; i < rows; i++) {
        for(j = 0; j < cols; j++) {
            printf("%d ", matrix[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

The code implements a simple matrix input and output program. It prompts the user for the number of rows and columns, then asks for matrix elements row by row. Finally, it prints the matrix row by row.

Below the code editor, the terminal window shows the execution of the program:

```
abhaygupta@192 main.c % gcc main.c
abhaygupta@192 main.c % ./a.out
Enter number of rows and columns: 2
2
Enter elements of the matrix:
22
12
34
11
Matrix is:
22 12
34 11
abhaygupta@192 main.c %
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The left sidebar (Explorer) lists files: .vscode, main.dSYM, a.out, hello.c, main, and main.c (which is currently selected). The center is a code editor with the file main.c open, displaying the following C code:

```
#include <stdio.h>
int main() {
    int rows, cols, i, j, sum = 0;
    printf("Enter number of rows and columns: ");
    scanf("%d %d", &rows, &cols);
    int matrix[rows][cols];
    printf("Enter elements of the matrix:\n");
    for(i = 0; i < rows; i++) {
        for(j = 0; j < cols; j++) {
            scanf("%d", &matrix[i][j]);
            sum += matrix[i][j];
        }
    }
    printf("Sum of all elements = %d\n", sum);
    return 0;
}
```

The right side shows a terminal window with the following session:

```
abhaygupta@192 main.c % gcc main.c
abhaygupta@192 main.c % ./a.out
Enter number of rows and columns: 2
Enter elements of the matrix:
1
2
3
4
Sum of all elements = 10
abhaygupta@192 main.c %
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The left sidebar displays the file tree (EXPLORER), showing files like main.c, .vscode, and various logs. The main area shows a code editor for 'main.c' with the following content:

```
#include <stdio.h>
int main() {
    int rows, cols, i, j;
    printf("Enter number of rows and columns: ");
    scanf("%d %d", &rows, &cols);
    int matrix[rows][cols], rowSum[rows];
    printf("Enter elements of the matrix:\n");
    for(i = 0; i < rows; i++) {
        for(j = 0; j < cols; j++)
            scanf("%d", &matrix[i][j]);
    }
    for(i = 0; i < rows; i++) {
        rowSum[i] = 0;
        for(j = 0; j < cols; j++)
            rowSum[i] += matrix[i][j];
    }
}
```

The terminal below shows the execution of the program and its output:

```
abhaygupta@192 main.c % ./a.out
Enter number of rows and columns: 3
Enter elements of the matrix:
1
2
3
4
5
6
7
8
9
Sum of each row:
Row 1 sum = 6
Row 2 sum = 15
Row 3 sum = 24
abhaygupta@192 main.c %
```

At the bottom, status bar details are visible: Ln 24, Col 5, Spaces: 4, UTF-8, LF, (C), Mac, Prettier.

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The left sidebar contains icons for Explorer, Search, Open Editors, and others. The main area has tabs for 'main.c' and 'main.c' (active). The code editor displays a C program for calculating the transpose of a matrix. The terminal at the bottom shows the execution of the program, including user input for rows and columns, and the resulting transpose output.

```
#include <stdio.h>
int main() {
    int rows, cols, i, j;
    printf("Enter number of rows and columns: ");
    scanf("%d %d", &rows, &cols);
    int matrix[rows][cols], transpose[cols][rows];
    printf("Enter elements of the matrix:\n");
    for(i = 0; i < rows; i++) {
        for(j = 0; j < cols; j++)
            scanf("%d", &matrix[i][j]);
    }
    for(i = 0; i < rows; i++) {
        for(j = 0; j < cols; j++)
            transpose[j][i] = matrix[i][j];
    }
    printf("Transpose of the matrix:\n");
    for(i = 0; i < cols; i++) {
        for(j = 0; j < rows; j++)
            printf("%d ", transpose[j][i]);
        printf("\n");
    }
}
```

TERMINAL

```
abhaygupta@192 main.c % gcc main.c
abhaygupta@192 main.c % ./a.out
Enter number of rows and columns: 2
2
Enter elements of the matrix:
12
23
45
67
Transpose of the matrix:
12 45
23 67
abhaygupta@192 main.c %
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The main area displays a C language file named `main.c`. The code implements a function to add two matrices. It prompts the user for the number of rows and columns, then for the elements of both matrices, and finally prints the sum of the two matrices.

```
#include <stdio.h>
int main() {
    int rows, cols, i, j;
    printf("Enter number of rows and columns: ");
    scanf("%d %d", &rows, &cols);
    int A[rows][cols], B[rows][cols], Sum[rows][cols];
    printf("Enter elements of first matrix:\n");
    for(i = 0; i < rows; i++)
        for(j = 0; j < cols; j++)
            scanf("%d", &A[i][j]);
    printf("Enter elements of second matrix:\n");
    for(i = 0; i < rows; i++)
        for(j = 0; j < cols; j++)
            scanf("%d", &B[i][j]);
    for(i = 0; i < rows; i++)
        for(j = 0; j < cols; j++)
            Sum[i][j] = A[i][j] + B[i][j];
}
```

The terminal tab at the bottom shows the execution of the code and its output:

```
abhaygupta@192 main.c % gcc main.c
abhaygupta@192 main.c % ./a.out
Enter number of rows and columns: 2
2
Enter elements of first matrix:
1
2
3
4
Enter elements of second matrix:
4
3
2
1
Sum of the two matrices:
5 5
5 5
abhaygupta@192 main.c %
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The left sidebar displays the 'EXPLORER' view with a tree structure of files. In the main area, a code editor window titled 'main.c M X' is open, showing the following C code:

```
#include <stdio.h>

int main() {
    int n, i, j, flag = 1;

    printf("Enter size of square matrix: ");
    scanf("%d", &n);

    int matrix[n][n];

    printf("Enter elements of the matrix:\n");
    for(i = 0; i < n; i++) {
        for(j = 0; j < n; j++) {
            scanf("%d", &matrix[i][j]);
        }
    }

    for(i = 0; i < n; i++) {
        for(j = 0; j < n; j++) {
            if(matrix[i][j] != matrix[j][i]) {
                flag = 0;
                break;
            }
        }
    }

    if(flag)
        printf("The matrix is symmetric.\n");
    else
        printf("The matrix is not symmetric.\n");
}

return 0;
```

The code implements a function to check if a square matrix is symmetric. It prompts the user for the size of the matrix and then for its elements. It then iterates through the matrix to compare each element with its transpose. If any element is found to be unequal, the 'flag' variable is set to 0 and the loop is broken. Finally, it prints whether the matrix is symmetric or not based on the value of 'flag'.

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The left sidebar displays the 'EXPLORER' view with several files listed under 'OPEN EDITORS' and 'MAIN.C'. The main editor area shows a C program named 'main.c'. The code prompts the user to enter the size of a square matrix and its elements. It then iterates through the matrix to check if all diagonal elements are distinct. If any two diagonal elements are found to be equal, the program sets a flag to 0 and breaks out of the inner loop. Finally, it prints a message indicating whether all diagonal elements are distinct or not.

```
C main.c M X
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4     int n, i, j, flag = 1;
5
6     printf("Enter size of square matrix: ");
7     scanf("%d", &n);
8
9     int matrix[n][n];
10    printf("Enter elements of the matrix:\n");
11    for(i = 0; i < n; i++) {
12        for(j = 0; j < n; j++) {
13            scanf("%d", &matrix[i][j]);
14
15        for(i = 0; i < n; i++) {
16            for(j = i + 1; j < n; j++) {
17                if(matrix[i][i] == matrix[j][j]) {
18                    flag = 0;
19                    break;
20                }
21            }
22            if(flag == 0)
23                break;
24        }
25
26        if(flag)
27            printf("All diagonal elements are distinct.\n");
28        else
29            printf("Diagonal elements are not distinct.\n");
30    }
}

```

The terminal output shows the result of running the program:

```
3
4
5
6
7
8
1
Diagonal elements are not distinct.
abhaygupta@192 main.c %
```

At the bottom, status bar details include: Ln 14, Col 1, Spaces: 4, UTF-8, LF, () C, Mac, Prettier.

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. On the left, the Explorer sidebar displays a file tree with several files: .vscode, main.dSYM, #include <stdio.h>, a.out, hello.c, main, main.c (which is currently selected), sample.txt, tempCodeRunnerFile, and Untitled-1. The main editor area shows the code for main.c:

```
C main.c M X
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4     int n, i, j, sum = 0;
5
6     printf("Enter size of square matrix: ");
7     scanf("%d", &n);
8
9     int matrix[n][n];
10
11    printf("Enter elements of the matrix:\n");
12    for(i = 0; i < n; i++) {
13        for(j = 0; j < n; j++) {
14            scanf("%d", &matrix[i][j]);
15            if(i == j)
16                sum += matrix[i][j];
17        }
18    }
19
20    printf("Sum of main diagonal elements = %d\n", sum);
21
22    return 0;
23 }
```

Below the editor, the terminal window shows the execution of the program:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
abhaygupta@192 main.c % ./a.out
Enter size of square matrix: 2
Enter elements of the matrix:
1
2
3
4
Sum of main diagonal elements = 5
abhaygupta@192 main.c %
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The left sidebar contains icons for file operations like Open, Save, Find, and Run. The Explorer sidebar shows a file tree with files like `main.c`, `.vscode`, `main.dSYM`, `stdio.c`, `a.out`, `hello.c`, `main`, and `sample.txt`. The main editor area displays a C program named `main.c` with the following code:

```
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4     int rows, cols, i, j;
5
6     printf("Enter number of rows and columns: ");
7     scanf("%d %d", &rows, &cols);
8
9     int matrix[rows][cols];
10
11    printf("Enter elements of the matrix:\n");
12    for(i = 0; i < rows; i++)
13        for(j = 0; j < cols; j++)
14            scanf("%d", &matrix[i][j]);
15
16    printf("Diagonal traversal of the matrix:\n");
17
18    for(int k = 0; k < cols; k++) {
19        i = 0;
20        j = k;
21        while(i < rows && j >= 0)
22            printf("%d ", matrix[i++][j--]);
23        printf("\n");
24    }
25
26    for(int k = 1; k < rows; k++) {
27        i = k;
28        j = cols - 1;
29        while(i < rows && j >= 0)
30            printf("%d ", matrix[i++][j--]);
31    }
32}
```

The terminal at the bottom shows the output of the program:

```
8
9
Diagonal traversal of the matrix:
1
2 4
3 5 7
6 8
9
```

The status bar at the bottom indicates the current file is `main.c` and the line and column are 17, 1.

A screenshot of the Visual Studio Code (VS Code) interface. The title bar shows the file name "main.c". The left sidebar (Explorer) lists files: ".vscode", "main.dSYM", "#include <stdio.h>", "a.out", "hello.c", "main", "main.c" (the active file), "sample.txt", "tempCodeRunnerFile", and "Untitled-1". The main editor area displays the following C code:

```
#include <stdio.h>
int main() {
    int r1, c1, r2, c2, i, j, k;
    printf("Enter rows and columns of first matrix: ");
    scanf("%d %d", &r1, &c1);
    printf("Enter rows and columns of second matrix: ");
    scanf("%d %d", &r2, &c2);
    if(c1 != r2) {
        printf("Matrix multiplication not possible!\n");
        return 0;
    }
    int A[r1][c1], B[r2][c2], C[r1][c2];
    printf("Enter elements of first matrix:\n");
    for(i = 0; i < r1; i++)
        for(j = 0; j < c1; j++)
            scanf("%d", &A[i][j]);
    printf("Enter elements of second matrix:\n");
    for(i = 0; i < r2; i++)
        for(j = 0; j < c2; j++)
            scanf("%d", &B[i][j]);
    for(i = 0; i < r1; i++)
        for(j = 0; j < c2; j++)

```

The terminal below shows the output of the program:

```
Enter elements of second matrix:
1
2
3
4
Resultant matrix after multiplication:
7 10
15 22
```