

Пример: Събиране две матрици a и b в матрица sum
(двумерни масиви от цели числа)

```
#include <iostream>
using namespace std;

int main()
{
    int r, c, a[100][100], b[100][100], sum[100][100], i, j;
    cout << "Enter number of rows (between 1 and 100): ";
    cin >> r;

    cout << "Enter number of columns (between 1 and 100): ";
    cin >> c;
    cout << endl << "Enter elements of 1st matrix: " << endl;

    // въвеждане на елементите за матрица, масив a
    for(i = 0; i < r; ++i)
        for(j = 0; j < c; ++j)
        {
            cout << "Enter element a" << i + 1 << j + 1 << " : ";
            cin >> a[i][j];
        }
    // въвеждане на елементите за матрица, масив b
    cout << endl << "Enter elements of 2nd matrix: " << endl;
    for(i = 0; i < r; ++i)
        for(j = 0; j < c; ++j)
        {
            cout << "Enter element b" << i + 1 << j + 1 << " : ";
            cin >> b[i][j];
        }
    // събиране на две матрици
    for(i = 0; i < r; ++i)
        for(j = 0; j < c; ++j)
            sum[i][j] = a[i][j] + b[i][j];

    // отпечатване на резултата от събирането на двете матрици в матрица sum.
    cout << endl << "Sum of two matrix is: " << endl;
    for(i = 0; i < r; ++i)
        for(j = 0; j < c; ++j)
        {
            cout << sum[i][j] << " ";
            if(j == c - 1)
                cout << endl;
        }
    return 0;
}
```

Изход от изпълнението на програмата

Enter number of rows (between 1 and 100): 2
Enter number of columns (between 1 and 100): 2

Enter elements of 1st matrix:

Enter element a11: -4
Enter element a12: 5
Enter element a21: 6
Enter element a22: 8

Enter elements of 2nd matrix:

Enter element b11: 3
Enter element b12: -9
Enter element b21: 7
Enter element b22: 2

Sum of two matrix is:

-1 -4
13 6

Пример: Транспониране на матрица (обръщане на редовете в колони и колоните в редове)

```
#include <iostream>
using namespace std;
int main()
{
    int a[10][10], trans[10][10], r, c, i, j;

    cout << "Enter rows and columns of matrix: ";
    cin >> r >> c;
    // въвеждане на елементите за масива, матрица a[][].
    cout << endl << "Enter elements of matrix: " << endl;
    for(i = 0; i < r; ++i)
        for(j = 0; j < c; ++j)
        {
            cout << "Enter elements a" << i + 1 << j + 1 << ": ";
            cin >> a[i][j];
        }
    // отпечатване на a[][]
    cout << endl << "Entered Matrix: " << endl;
    for(i = 0; i < r; ++i)
        for(j = 0; j < c; ++j)
        {
            cout << " " << a[i][j];
            if(j == c - 1)
                cout << endl << endl;
        }
}
```

```

// транспониране на a[][] и записване в trans[][].
for(i = 0; i < r; ++i)
    for(j = 0; j < c; ++j)
    {
        trans[j][i]=a[i][j];
    }

// отпечатване на trans[][].
cout << endl << "Transpose of Matrix: " << endl;
for(i = 0; i < c; ++i)
    for(j = 0; j < r; ++j)
    {
        cout << " " << trans[i][j];
        if(j == r - 1)
            cout << endl << endl;
    }
return 0;
}

```

Изход от изпълнението на програмата

Enter rows and column of matrix: 2
3

Enter elements of matrix:
Enter elements a11: 1
Enter elements a12: 2
Enter elements a13: 9
Enter elements a21: 0
Enter elements a22: 4
Enter elements a23: 7

Entered Matrix:
1 2 9

0 4 7

Transpose of Matrix:
1 0
2 4
9 7

Пример: Предаване на двумерен масив като параметър към функция

```
#include <iostream>
#define M 3
#define N 2

using namespace std;

void display(int n[M][N]);

int main()
{
    int num[M][N] = {
        {3, 4},
        {9, 5},
        {7, 1}
    };

    display(num);

    return 0;
}

void display(int n[M][N])
{
    cout << "Displaying Values: " << endl;
    for(int i = 0; i < M; ++i)
    {
        for(int j = 0; j < N; ++j)
        {
            cout << n[i][j] << " ";
        }
        cout << endl;
    }
}
```

Изход от изпълнението на програмата

```
Displaying Values:
3 4
9 5
7 1
```

Пример: попълване със случайни стойности на масив

```
#include <iostream>
#include <cstdlib>
#include <ctime>
#include <iomanip>
using namespace std;
//инициализиране на масив със случайно генерирани числа в инт. [-5, 15]
void fillArray(int ** arr, unsigned n, unsigned m) {
    srand(time(NULL));
    for(unsigned i=0; i<n; i++){
        for(unsigned j=0; j<m; j++){
            arr[i][j] = rand() % 20 -5;
        }
    }
}

void printArray(int ** arr, unsigned n, unsigned m){
    for(unsigned i=0; i<n; i++){
        cout << "{";
        for(unsigned j=0; j<m; j++){
            cout << setw(3) << arr[i][j];
            if(j<m-1) cout << ",";
        }
        cout << " }";
        cout << endl;
    }
}

int main() {
    const unsigned N = 4;
    const unsigned M = 5;

    //заделяне на област от паметта
    int ** matrix = new int*[N];
    for(unsigned i=0; i<N; i++){
        matrix[i] = new int[M];
    }
    cout << "Generate matrix:" << endl;
    fillArray(matrix, N, M);
    cout << "Print Matrix:" << endl;
    printArray(matrix, N, M);

    //освобождаване на заделената памет
    for(unsigned i=0; i<N; i++){
        delete matrix[i];
    }
    delete [] matrix;

    return 0;
}
```

Символни низове

C-style string

Пример: Премахване на всички знаци с изключение на буквите

Тази програма чете низ (C-style string) от потребителя и премахва всички знаци с изключение на буквите.

```
#include <iostream>
using namespace std;

int main() {
    char line[100], alphabetString[100];
    int j = 0;
    cout << "Enter a string: ";
    cin.getline(line, 100);

    for(int i = 0; line[i] != '\0'; ++i)
    {
        if ((line[i] >= 'a' && line[i] <= 'z') || (line[i] >= 'A' && line[i] <= 'Z'))
        {
            alphabetString[j++] = line[i];
        }
    }
    alphabetString[j] = '\0';

    cout << "Output String: " << alphabetString;
    return 0;
}
```

Изход от изпълнението на програмата

```
Enter a string: CPP P2'r"o@gram84_ZX./
Output String: CPP ProgramZX
```

Пример: Конкатенация (“слепване”) на два низа в един

```
#include <iostream>
#include <cstring>

using namespace std;

int main()
{
    char s1[50], s2[50], result[100];

    cout << "Enter string s1: ";
    cin.getline(s1, 50);

    cout << "Enter string s2: ";
    cin.getline(s2, 50);

    strcat(s1, s2);

    cout << "s1 = " << s1 << endl;
    cout << "s2 = " << s2;

    return 0;
}
```