```cpp
#include <iostream>
#include <string>
#include <fstream>
#include <iomanip>
using namespace std;

void get_command(string& command);        // 10점
void get_size(int& size);                 // 10점
int** gen_matrix(int size);                         // 20점
void swap(int* a, int* b);                          // 10점
void sort_array(int* ary, int size); // 10점
void sort_matrix_row(int** matrix, int size);       // 10점
void print_matrix(int** matrix, int size);          // 10점
void save_matrix(int** matrix, int size);           //10점
void free_matrix(int** matrix, int size);           // 10점

int main() {
        string command;
        int** matrix = NULL;
        int size = 0;
        while (1) {
                get_command(command); // 1)

                if(command == "1"){
                        get_size(size); //2)
                        matrix = gen_matrix(size); //3)
                }else if (command == "2") {
                        print_matrix(matrix, size);
                }else if (command == "3") {
                        sort_matrix_row(matrix, size);
                }else if (command == "4") {
                        save_matrix(matrix, size);
                }else if (command == "0") {
                        free_matrix(matrix, size);
                        cout << "Exit the program.." << endl;
                        exit(104);
                }else {
                        cout << "Wrong command" << endl;
                }
        }

        return 0;
}


void get_command(string& command) {
        cout << "1. Generate matrix" << endl;
        cout << "2. Print matrix" << endl;
        cout << "3. Sort matrix" << endl;
        cout << "4. Save matrix" << endl;
        cout << "0. Exit program" << endl;
```

```
        cout << ">>";
        cin >> command;
}

void get_size(int& size) {
        do {
                cout << "Enter the size in (size x size): ";
                cin >> size;
        } while (size < 2);
}

int** gen_matrix(int size) {
        int**matrix = new int*[size];
        for (int i = 0; i < size; i++)
                matrix[i] = new int[size];

        for (int row = 0; row < size; row++) {
                for (int col = 0; col < size; col++) {
                        matrix[row][col] = (rand()) % 101;
                }
        }
        cout << endl;

        return matrix;
}

void swap(int* a, int* b) {
        int temp = *a;
        *a = *b;
        *b = temp;
}

void sort_array(int* ary, int size) {
        for (int i = 0; i < (size - 1); i++) {
                for (int j = i; j < (size - 1); j++) {
                        if (ary[j] > ary[j + 1])
                                swap(ary[j], ary[j + 1]);
                }
        }
}

void sort_matrix_row(int** matrix, int size) {

        for (int row = 0; row < size; row++) {
                sort_array(matrix[row], size);
        }
        cout << "Completed" << endl;
}

void print_matrix(int** matrix, int size) {
        for (int row = 0; row < size; row++) {
```

```
                for (int col = 0; col < size; col++) {
                        cout << setw(4) << matrix[row][col];
                }
                cout << endl;
        }
        cout << endl;
}

void save_matrix(int** matrix, int size) {
        ofstream fout("matrix.txt");
        fout << size << endl;
        for (int row = 0; row < size; row++) {
                for (int col = 0; col < size; col++) {
                        fout << setw(4) << matrix[row][col];
                }
                fout << endl;
        }
        fout.close();

        cout << "Saved" << endl;
        cout << endl;
}

void free_matrix(int** matrix, int size) {
        for (int i = 0; i < size; i++)
                delete[] matrix[i];
        delete[] matrix;
}
```