// Как для одномерного так и для двумерного массива

// СПОСОБЫ ЗАПИСИ МАССИВА В ФУНКЦИЮ В КАЧЕСТВЕ АРГУМЕНТА //

#include <iostream>

using namespace std;

//указатель на массив//

void one\_in(int massiv[], const int razmer)

{

for (int i = 0; i < razmer; i++)

{

massiv[i] = rand () % 10;

}

}

void one\_out(int massiv[], const int razmer)

{

for (int i = 0; i < razmer; i++)

{

cout << massiv[i] << " ";

}

}

void dv\_one\_in(int dv\_massiv[][3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

dv\_massiv[i][j] = rand() % 10;

}

}

}

void dv\_one\_out(int dv\_massiv[][3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

cout << dv\_massiv[i][j] << "\t";

}

cout << endl;

}

}

//ссылка на массив//

void two\_in(int(&massiv)[9], const int razmer)

{

for (int i = 0; i < razmer; i++)

{

massiv[i] = rand () % 10;

}

}

void two\_out(int(&massiv)[9], const int razmer)

{

for (int i = 0; i < razmer; i++)

{

cout << massiv[i] << " ";

}

}

void dv\_two\_in(int (&dv\_massiv)[3][3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

dv\_massiv[i][j] = rand() % 10;

}

}

}

void dv\_two\_out(int (&dv\_massiv)[3][3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

cout << dv\_massiv[i][j] << "\t";

}

cout << endl;

}

}

//указатель на ссылку//

void three\_in(int massiv[], const int razmer)

{

int& ykazat = massiv[0];

int\* ssilk = &ykazat;

for (int i = 0; i < razmer; i++)

{

ssilk[i] = rand () % 10;

}

}

void three\_out(int massiv[], const int razmer)

{

int& ykazat = massiv[0];

int\* ssilk = &ykazat;

for (int i = 0; i < razmer; i++)

{

cout << ssilk[i] << " ";

}

}

void dv\_three\_in(int dv\_massiv[][3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

dv\_massiv[i][j] = rand() % 10;

}

}

}

void dv\_three\_out(int dv\_massiv[][3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

cout << dv\_massiv[i][j] << "\t";

}

cout << endl;

}

}

//ссылка на указатель//

void four\_in(int& massiv, const int razmer)

{

int\* mass = &massiv;

for (int i = 0; i < razmer; i++)

{

mass[i] = rand () % 10;

}

}

void four\_out(int& massiv, const int razmer)

{

int\* mass = &massiv;

for (int i = 0; i < razmer; i++)

{

cout << mass[i] << " ";

}

}

void dv\_four\_in(int (&dv\_m)[3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

(&dv\_m[i])[j] = rand() % 10;

}

}

}

void dv\_four\_out(int (&dv\_m)[3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

cout << (&dv\_m[i])[j] << "\t";

}

cout << endl;

}

}

//указатель на указатель//

void five\_in(int massiv[], const int razmer)

{

for (int i = 0; i < razmer; i++)

{

massiv[i] = rand () % 10;

}

}

void five\_out(int massiv[], const int razmer)

{

for (int i = 0; i < razmer; i++)

{

cout << massiv[i] << " ";

}

}

void dv\_five\_in(int (\*dv\_ma)[3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

dv\_ma[i][j] = rand() % 10;

}

}

}

void dv\_five\_out(int (\*dv\_ma)[3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

cout << dv\_ma[i][j] << "\t";

}

cout << endl;

}

}

//ссылка на ссылку//

void six\_in(int& massiv, const int razmer)

{

int& yk1 = massiv;

int\* ms = &yk1;

for (int i = 0; i < razmer; i++)

{

ms[i] = rand () % 10;

}

}

void six\_out(int& massiv, const int razmer)

{

int& yk1 = massiv;

int\* ms = &yk1;

for (int i = 0; i < razmer; i++)

{

cout << ms[i] << " ";

}

}

void dv\_six\_in(int (&dv\_ma)[3][3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

dv\_ma[i][j] = rand() % 10;

}

}

}

void dv\_six\_out(int (&dv\_massiv)[3][3], const int lin, const int col)

{

for (int i = 0; i < lin; i++)

{

for (int j = 0; j < col; j++)

{

cout << dv\_massiv[i][j] << "\t";

}

cout << endl;

}

}

int main()

{

setlocale(LC\_ALL, "Russian");

const int razmer = 9;

int massiv[razmer];

const int lin = 3;

const int col = 3;

int dv\_massiv [lin][col];

cout << endl << "Первый способ - указатель на массив: \n" << endl << "Одномерный массив: " << endl;

one\_in(massiv, razmer);

one\_out(massiv, razmer);

cout << endl << endl << "Двумерный массив: " << endl;

dv\_one\_in(dv\_massiv, lin, col);

dv\_one\_out(dv\_massiv, lin, col);

cout << endl << endl << endl << "Второй способ - ссылка на массив: \n" << endl << "Одномерный массив: " << endl;

two\_in(massiv, razmer);

two\_out(massiv, razmer);

cout << endl << endl << "Двумерный массив: " << endl;

dv\_two\_in(dv\_massiv, lin, col);

dv\_two\_out(dv\_massiv, lin, col);

cout << endl << endl << endl << "Третий способ - указатель на ссылку: \n" << endl << "Одномерный массив: " << endl;

int& ykazat = massiv[0];

int\* ssilk = &ykazat;

three\_in(ssilk, razmer);

three\_out(ssilk, razmer);

cout << endl << endl << "Двумерный массив: " << endl;

int (&dv\_yk)[3][3] = dv\_massiv;

int (\*dv\_ss)[3] = dv\_yk;

dv\_three\_in(dv\_ss, lin, col);

dv\_three\_out(dv\_ss, lin, col);

cout << endl << endl << endl << "Четвёртый способ - ссылка на указатель: \n" << endl << "Одномерный массив: " << endl;

int\* mas\_yk = massiv;

int& mas\_ss = \*mas\_yk;

four\_in(mas\_ss, razmer);

four\_out(mas\_ss, razmer);

cout << endl << endl << "Двумерный массив: " << endl;

int (\*dv\_uk)[3] = dv\_massiv;

dv\_four\_in(\*dv\_uk, lin, col);

dv\_four\_out(\*dv\_uk, lin, col);

cout << endl << endl << endl << "Пятый способ - указатель на указатель: \n" << endl << "Одномерный массив: " << endl;

int\* mas\_ssilk1 = massiv;

int\* mas\_ssilk2 = mas\_ssilk1;

five\_in(mas\_ssilk2, razmer);

five\_out(mas\_ssilk2, razmer);

cout << endl << endl << "Двумерный массив: " << endl;

int (\*d\_m)[3] = dv\_massiv;

dv\_five\_in(d\_m, lin, col);

dv\_five\_out(d\_m, lin, col);

cout << endl << endl << endl << "Шестой способ - ссылка на ссылку: \n" << endl << "Одномерный массив: " << endl;

int& yk1 = massiv[0];

int& yk2 = yk1;

six\_in(yk2, razmer);

six\_out(yk2, razmer);

cout << endl << endl << "Двумерный массив: " << endl;

int (&d\_ma)[3][3] = dv\_massiv;

dv\_six\_in(d\_ma, lin, col);

dv\_six\_out(d\_ma, lin, col);

cout << endl;

}