“В двумерном массиве хранится информация о количестве студентов в той или иной группе каждого курса института с первого по пятый (в первой строке – информация о группах первого курса, во второй – второго и т.д.). На каждом курсе имеется 8 групп”

Оглавление

[Задача 2](#_Toc438563486)

[Описательный алгоритм решения задачи 2](#_Toc438563487)

[Блочный алгоритм 3](#_Toc438563488)

[Функция ввода массива (InArr) 3](#_Toc438563489)

[Функция вывода массива(OutArr) 4](#_Toc438563490)

[Функция нахождения суммы (Sum) 5](#_Toc438563491)

[Тело программы (main) 6](#_Toc438563492)

[Код программы 7](#_Toc438563493)

# Задача

Целью данной работы является знакомство с средствами Microsoft Office Word.

Составьте описательный и блочный алгоритм решения задачи. Написать программу на языке С++ для расчета общего числа студентов на любом курсе. Ввод, вывод элементов массива, а также требуемые действия реализовать с помощью функций. Все функции должны иметь независимый интерфейс.

**Входные данные:** Двумерный массив количества студентов в каждой группе каждого курса. Число n равное номеру курса.

**Выходные данные:** Число, равное общему количеству студентов на n-ом курсе.

# Описательный алгоритм решения задачи

1. Объявить константные переменные group и course;
2. Объявить массив students;
3. Заполнить массив случайными числами;
4. Вывести массив;
5. Ждать ввода номера группы – значения curr;
6. Считать сумму (curr-1) строчки массива;
7. Вывести результат;

# Блочный алгоритм

## Функция ввода массива (InArr)

InArr(\*students, course, group)

Int i=0, j=0

От i=0  
До i=course  
i++

От j=0  
До j=group  
j++

\*(student + j + i\*course) = rand() % 30+5

Конец

## Функция вывода массива(OutArr)

Int i=0, j=0

От i=0  
До i=course  
i++

endl

\*(student + j + i\*course)  
\t

Конец

От j=0  
До j=group  
j++

OutArr(\*students, course, group)

## Функция нахождения суммы (Sum)

ss+=\*(students + curr + i);

int i, ss=0;

От i=0  
До i=group  
i++

Sum(\*students, curr, group)

return(ss);

Конец

## Тело программы (main)

const int group=8, course=5;  
int students [course][group], curr;

Конец

sum(\*students, curr\*course, group)

curr--;

"The number of students in", curr, " course= "

curr

"Enter the course number"

OutArr(\*students, course, group)

"The number of students in each groups"

InArr(\*students, course, group)

Srand(time(NULL))

main()

# Код программы

#include "stdafx.h"

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;[[1]](#endnote-1)

void[[2]](#endnote-2) InArr(int \*students, int course, int group) [[3]](#footnote-1)

{

int[[4]](#endnote-3) i, j;

for[[5]](#endnote-4) (i = 0; i < course; i++)

for (j = 0;j < group; j++)

{

\*(students + j + i\*course) = rand[[6]](#endnote-5)() % 30+5;

}

}

void OutArr(int \*students, int course, int group)[[7]](#footnote-2)

{

int i, j;

for (i = 0; i < course; i++) {

for (j = 0;j < group; j++)

{

cout[[8]](#endnote-6) << \*(students + j + i\*course) << "\t";

}

cout << endl;

}

}

int sum(int \*students, int curr, int group)[[9]](#footnote-3)

{

int i, ss=0;

for (i = 0; i < group; i++)

{

ss += \*(students + curr + i);

}

return[[10]](#endnote-7)(ss);

}

void main()[[11]](#footnote-4)

{

const[[12]](#endnote-8) int group = 8;

const int course = 5;

int students[course][group] = {}, curr;

srand(time(NULL[[13]](#endnote-9)));

InArr(\*students, course, group);

cout << "The number of students in each groups" << endl;

OutArr(\*students, course, group);

cout << "Enter the course number" << endl;

cin[[14]](#endnote-10) >> curr;

cout << "The number of students in " << curr << " course= ";

curr--;[[15]](#footnote-5)

cout<<sum(\*students, curr\*course, group) << endl;[[16]](#footnote-6)

}

1. Использование стандартного потока данных [↑](#endnote-ref-1)
2. Функция не возвращающая значение [↑](#endnote-ref-2)
3. //Функция ввода массива [↑](#footnote-ref-1)
4. Целочисленный тип данных [↑](#endnote-ref-3)
5. Цикл For [↑](#endnote-ref-4)
6. Генератор случайных чисел [↑](#endnote-ref-5)
7. //Функция вывода массива [↑](#footnote-ref-2)
8. Вывод значения в стандартный поток данных [↑](#endnote-ref-6)
9. //Поиск суммы строки массива [↑](#footnote-ref-3)
10. Возвращает значение переменной ss [↑](#endnote-ref-7)
11. //Тело программы [↑](#footnote-ref-4)
12. Константное значение [↑](#endnote-ref-8)
13. Инициализация нулем [↑](#endnote-ref-9)
14. Считывание из стандартного потока данных [↑](#endnote-ref-10)
15. //т.к. номера строк в массиве на единицу меньше номера курса [↑](#footnote-ref-5)
16. //curr\*course предопределяет строку в которой ведется подсчет [↑](#footnote-ref-6)