МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет прикладной математики, информатики и механики Кафедра математического обеспечения ЭВМ

## КУРСОВАЯ РАБОТА

по дисциплине «Информатика и программирование»

*Направление 02.03.02 – Фундаментальная информатика и информационные технологии*

*Профиль – Инженерия программного обеспечения (ФГОС3++)*

Зав. Кафедрой доктор техн. наук, проф. Ф.И. Зав Кафедпрй

Обучающийся, студент 1 курса Ф.И. Студент

Руководитель, канд. техн. наук, доцент Ф.И.Препод

### Город Год

### Содержание

[Постановка задачи 3](#_bookmark0)

[Описание данных и алгоритма решения задачи 4](#_bookmark1)

[Описание структуры программы 17](#_bookmark2)

[Результаты тестирования 18](#_bookmark3)

[Список использованных источников 19](#_bookmark4)

Приложение 20

# Постановка задачи

С клавиатуры вводится информация об итогах первой экзаменационной сессии студентов 1 курса 62 группы факультета ПММ ВГУ. Эта информация включает:

* целое число n (количество студентов);
* одномерный массив, содержащий их фамилии и имена;
* прямоугольную таблицу А (результаты экзаменов первой сессии), каждая строка которой содержит пять целых чисел (0,2,3,4,5 (0 – неявка на экзамен, остальные цифры соответствуют оценкам: 2 – неудовлетворительно, 3 – удовлетворительно, 4 – хорошо, 5 – отлично))
* прямоугольную таблицу Z, каждая строка которой, содержит результаты сдачи k зачётов (+ зачёт сдан, - зачёт не сдан)

Напечатать фамилии и имена, оценки тех студентов, средний балл успеваемости (по результатам экзаменов) которых ниже среднего балла отличников и хорошистов.

Технические требования:

При выполнении заданий курсовой работы необходимо использовать функции, адреса и указатели, сохранять в памяти компьютера исходную матрицу без изменений, чтобы напечатать её в конце программы.

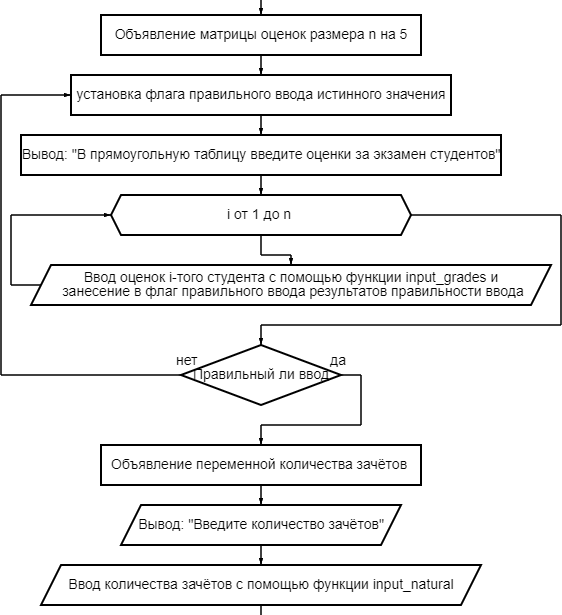
# Описание данных и алгоритма решения задачи

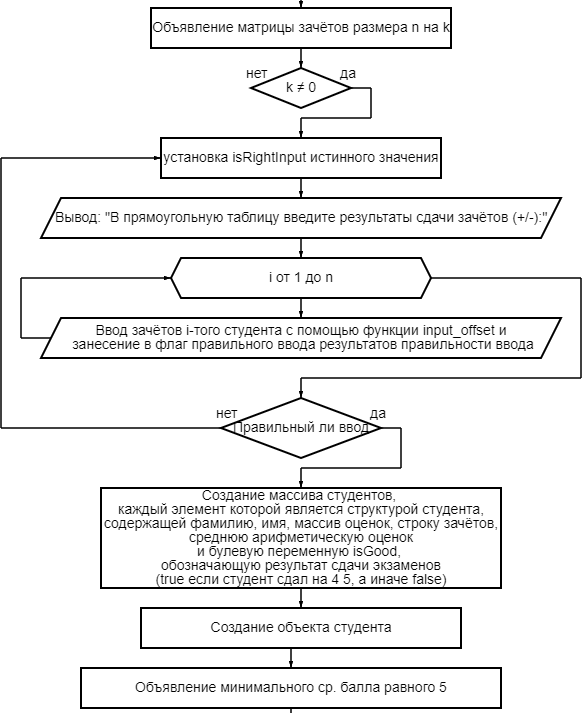
Для реализации решения задачи с использованием динамических массивов использовались следующие простые и структурированные данные: количество студентов n, количество зачётов k, двумерные массивы A и Z (оценки студентов и их зачёты соответственно), структура Student, содержащая фамилию surname и имя name студента типа string, целочисленный массив marks оценок студента, зачёты студента в строковом формате string, среднюю арифметическую оценок averageMark типа flaot и переменную isGood булевского типа обозначающую успешную сдачу студентов сессии, массив данных структур allStudents и объект структуры student, а также переменную булевского типа isRightInput – флаг правильности вводимых пользователем данных. В программе используются функции: input\_natural(ввод целого числа в диапозоне), input\_grades(ввод оценок студента в массив), input\_offset(ввод зачётов студента в массив), input\_student(передача в структуру Student данных о студенте), output\_student(вывод данных структуры Student), average\_mark(нахождение среднего арифметического оценок), is\_marks\_good(выяснение сдал ли студент сессию успешно(оценки за экзамен только 4 и 5)) . После выполнения всех этих операций, программа выводит на экран имена, фамилии и оценки студентов, у которых

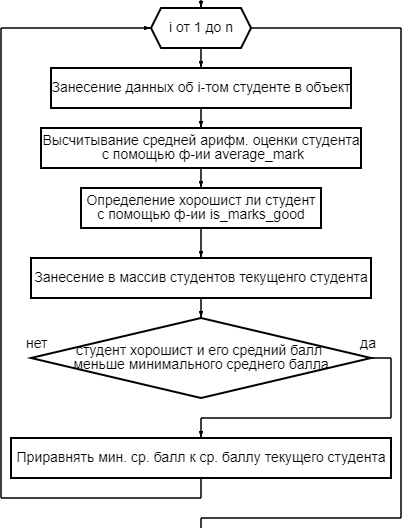
Для наглядности представим алгоритм решения поставленной задачи в виде блок-схемы для основной части задания и блок-схем для функций, используемых в программе.

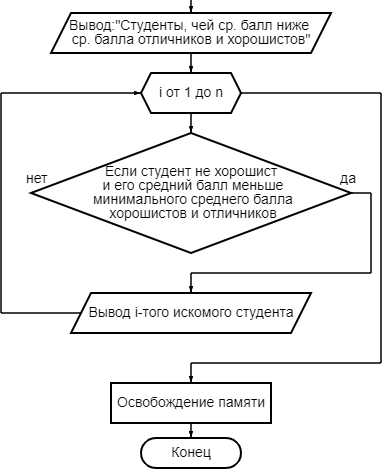
### Общее описание алгоритма решения задачи



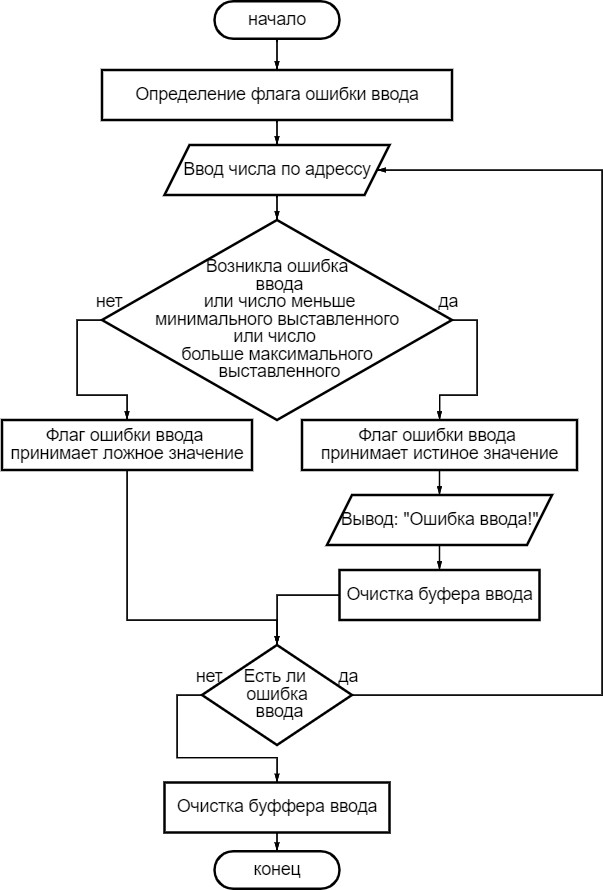






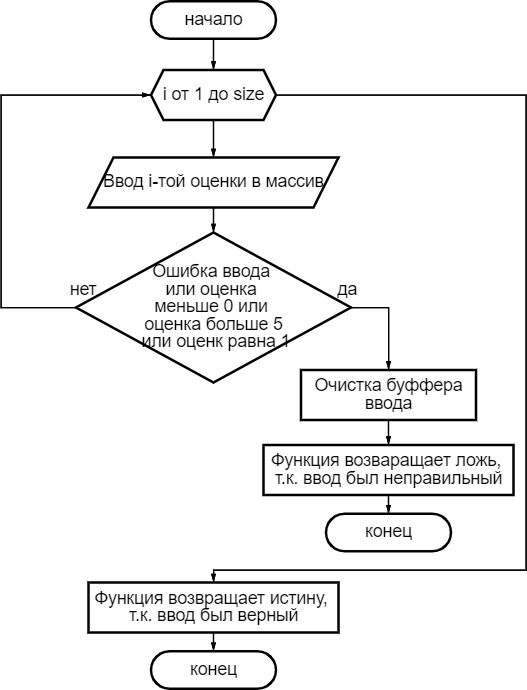


#### Описание алгоритма функции для ввода целого числа диапазоне **input\_natural**

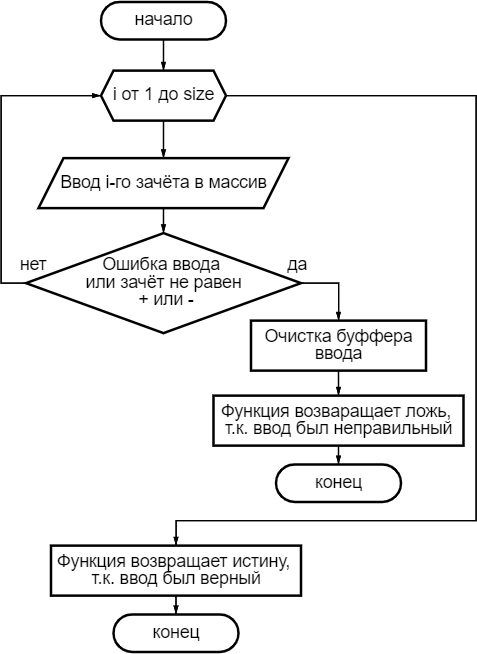


#### Описание алгоритма функции для ввода оценок за экзамен студента

### input\_grades



#### Описание алгоритма функции для ввода зачётов студента



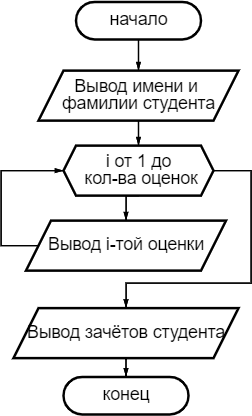
### input\_offset

#### Описание алгоритма функции для ввода данных о студенте в структуру Student **input\_sutdent**

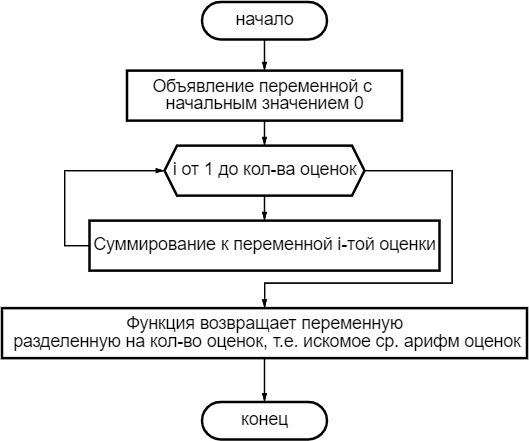


#### Описание алгоритма функции вывод структуры студента

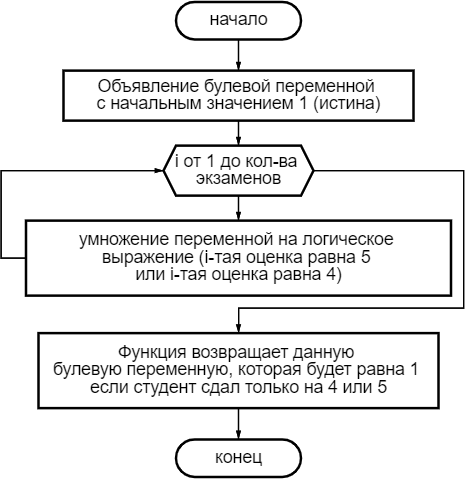
### output\_student



#### Описание алгоритма функции для подсчёта среднего арифметического балла **average\_mark**



#### Описание алгоритма функции для выяснения сдал ли студент сессию успешно (т.е. на оценки 4 и 5) **is\_marks\_good**



# Описание структуры программы

#### Просим пользователя ввести число студентов до тех пор, пока не будет введено корректное целое положительное число.

#### Резервируем память для входных данных под введённое на шаге 1 количество студентов. Просим пользователя ввести информацию о студентах до тех пор, пока она не будет

#### соответствовать шаблону текста задачи.

#### Резервируем память для массива структур под введённое на шаге 1 количество студентов. Определяем средний балл и успешность сдачи экзаменов для каждого студента.

#### Выводим на экран имя, фамилию и оценки тех студентов, средний балл успеваемости (по результатам экзаменов)

#### которых ниже среднего балла отличников и хорошистов.

#### Высвобождаем резервированную память.

# Результаты тестирования

#### Входные данные:

#### 3

#### Иванов Иван Петров Пётр Степанов Степан 4 4 5 4 4

#### 3 2 0 4 2

#### 4 5 5 5 5

#### 2

#### + -

#### + +

#### + +

#### Результаты выполнения программы:

#### Студенты, чей средний балл ниже среднего балла отличников и хорошистов:

#### Петров Пётр 3 2 0 4 2 + +

#### Входные данные:

#### 5

#### Полежайкин Илья Васнецова Полина Сергеева Галина Захаров Евгений Кожемятько Тамара

#### 2 2 0 3 4

#### 4 4 4 4 4

#### 5 5 5 5 5

#### 3 5 5 4 5

#### 0 4 3 2 2

#### 1

#### +

#### +

#### +

#### -

#### -

#### Результаты выполнения программы:

#### Студенты, чей средний балл ниже среднего балла отличников и хорошистов:

#### Полежайкин Илья 2 2 0 3 4 +

#### Кожемятько Тамара 0 4 3 2 2 -

# Список использованных источников

#### Вирт Н. Алгоритмы и структуры данных / Н. Вирт. – М.: Мир, 2016. -360

#### Структуры в языке C++ [Электронный ресурс]- https://ravesli.com/urok-61-struktury/