

**Перечень теоретических и практических заданий к экзамену  
по ОП.04 Основы алгоритмизации и программирования  
(2 курс, 4 семестр 2025-2026 уч. г.)**

**Форма контроля:** Практическая работа (Опрос)

**Описательная часть:** По выбору выполнить 1 теоретическое задание и 3 практических задания

**Перечень теоретических заданий:**

**Задание №1**

Ответить на тестовые вопросы, в личном кабинете ИАС "ИркАТ" – " ОП.04 Основы алгоритмизации и программирования. Экзамен".

Оценка	Показатели оценки
5	Даны верные ответы на 95% вопросов.
4	Даны верные ответы на 75% вопросов.
3	Даны верные ответы на 50% вопросов.

**Задание №2**

Вопрос 1:

Для решения любой задачи с помощью компьютера необходимо выполнить следующие этапы:

*Установите правильную последовательность этапов.*

Ответ

1	Постановка задачи
2	Математическое моделирование
3	Алгоритмизация задач
4	Программирование
5	Анализ результатов

Вопрос 2:

*Соотнесите свойства алгоритма с их описанием:*

Ответ

Результативность	алгоритм должен обеспечить выдачу результата решения задачи на печать, на экран монитора или в файл
Конечность	алгоритм должен приводить к решению задачи обязательно за конечное время
Эффективность	позволяет решить задачу за приемлемое для разработчика время
Массовость	правильный результат по алгоритму получен для одних исходных данных, то правильный результат по этому же алгоритму должен быть получен и для других исходных данных, допустимых в данной задаче
Определенность	неоднозначность толкования алгоритма недопустима

Вопрос 3:

Выберите тип алгоритма, описанного ниже:

*Набор команд (указаний), выполняемых последовательно друг за другом*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 4:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 5:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, предусматривающий многократное повторение одного и того же действия над новыми исходными данными*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Даны верные ответы на 1 вопрос.

### Задание №3

Вопрос 1:

Для решения любой задачи с помощью компьютера необходимо выполнить следующие этапы:

*Установите правильную последовательность этапов.*

Ответ

1	Постановка задачи
2	Математическое моделирование
3	Алгоритмизация задач
4	Программирование
5	Анализ результатов

Вопрос 2:

*Соотнесите свойства алгоритма с их описанием:*

Ответ

Результативность	алгоритм должен обеспечить выдачу результата решения задачи на печать, на экран
------------------	---------------------------------------------------------------------------------

Конечность	монитора или в файл алгоритм должен приводить к решению задачи обязательно за конечное время
Эффективность	позволяет решить задачу за приемлемое для разработчика время
Массовость	правильный результат по алгоритму получен для одних исходных данных, то правильный результат по этому же алгоритму должен быть получен и для других исходных данных, допустимых в данной задаче
Определенность	неоднозначность толкования алгоритма недопустима

Вопрос 3:

Выберите тип алгоритма, описанного ниже:

*Набор команд (указаний), выполняемых последовательно друг за другом*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 4:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 5:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, предусматривающий многократное повторение одного и того же действия над новыми исходными данными*

Ответ

- Линейный
- Разветвляющийся

1. Циклический

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

#### Задание №4

Вопрос 1:

Для решения любой задачи с помощью компьютера необходимо выполнить следующие этапы:

*Установите правильную последовательность этапов.*

Ответ

1	Постановка задачи
2	Математическое моделирование
3	Алгоритмизация задач
4	Программирование
5	Анализ результатов

Вопрос 2:

*Соотнесите свойства алгоритма с их описанием:*

Ответ

Результативность	алгоритм должен обеспечить выдачу результата решения задачи на печать, на экран монитора или в файл
------------------	-----------------------------------------------------------------------------------------------------

Конечность	алгоритм должен приводить к решению задачи обязательно за конечное время
Эффективность	позволяет решить задачу за приемлемое для разработчика время
Массовость	правильный результат по алгоритму получен для одних исходных данных, то правильный результат по этому же алгоритму должен быть получен и для других исходных данных, допустимых в данной задаче
Определенность	неоднозначность толкования алгоритма недопустима

Вопрос 3:

Выберите тип алгоритма, описанного ниже:

*Набор команд (указаний), выполняемых последовательно друг за другом*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 4:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 5:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, предусматривающий многократное повторение одного и того же действия над новыми исходными данными*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №5

Вопрос 1:

Для решения любой задачи с помощью компьютера необходимо выполнить следующие этапы:

*Установите правильную последовательность этапов.*

Ответ

1	Постановка задачи
2	Математическое моделирование
3	Алгоритмизация задач
4	Программирование
5	Анализ результатов

Вопрос 2:

*Соотнесите свойства алгоритма с их описанием:*

Ответ

Результативность	алгоритм должен обеспечить выдачу результата решения задачи на печать, на экран монитора или в файл
Конечность	алгоритм должен приводить к решению задачи обязательно за конечное время

Эффективность

позволяет решить задачу за приемлемое для разработчика время

Массовость

правильный результат по алгоритму получен для одних исходных данных, то правильный результат по этому же алгоритму должен быть получен и для других исходных данных, допустимых в данной задаче

Определенность

неоднозначность толкования алгоритма недопустима

Вопрос 3:

Выберите тип алгоритма, описанного ниже:

*Набор команд (указаний), выполняемых последовательно друг за другом*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 4:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 5:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, предусматривающий многократное повторение одного и того же действия над новыми исходными данными*



Ответ

- Линейный
- Разветвляющийся
- Циклический

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №6

Вопрос 1:

Для решения любой задачи с помощью компьютера необходимо выполнить следующие этапы:

*Установите правильную последовательность этапов.*

Ответ

1	Постановка задачи
2	Математическое моделирование
3	Алгоритмизация задач
4	Программирование
5	Анализ результатов

Вопрос 2:

*Соотнесите свойства алгоритма с их описанием:*

Ответ

Результативность	алгоритм должен обеспечить выдачу результата решения задачи на печать, на экран монитора или в файл
Конечность	алгоритм должен приводить к решению задачи обязательно за конечное время
Эффективность	позволяет решить задачу за приемлемое для разработчика время

Массовость

правильный результат по алгоритму получен для одних исходных данных, то правильный результат по этому же алгоритму должен быть получен и для других исходных данных, допустимых в данной задаче

Определенность

неоднозначность толкования алгоритма недопустима

Вопрос 3:

Выберите тип алгоритма, описанного ниже:

*Набор команд (указаний), выполняемых последовательно друг за другом*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 4:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 5:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, предусматривающий многократное повторение одного и того же действия над новыми исходными данными*

Ответ

- Линейный

- Разветвляющийся
- Циклический

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №7

Вопрос 1:

Для решения любой задачи с помощью компьютера необходимо выполнить следующие этапы:

*Установите правильную последовательность этапов.*

Ответ

1	Постановка задачи
2	Математическое моделирование
3	Алгоритмизация задач
4	Программирование
5	Анализ результатов

Вопрос 2:

*Соотнесите свойства алгоритма с их описанием:*

Ответ

Результативность	алгоритм должен обеспечить выдачу результата решения задачи на печать, на экран монитора или в файл
Конечность	алгоритм должен приводить к решению задачи обязательно за конечное время
Эффективность	позволяет решить задачу за приемлемое для разработчика время
Массовость	правильный результат по алгоритму получен для одних исходных данных, то

правильный результат по этому же алгоритму должен быть получен и для других исходных данных, допустимых в данной задаче

неоднозначность толкования алгоритма недопустима

Определенность

Вопрос 3:

Выберите тип алгоритма, описанного ниже:

*Набор команд (указаний), выполняемых последовательно друг за другом*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 4:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Вопрос 5:

Выберите тип алгоритма, описанного ниже:

*Алгоритм, предусматривающий многократное повторение одного и того же действия над новыми исходными данными*

Ответ

- Линейный
- Разветвляющийся
- Циклический

Оценка	Показатели оценки
--------	-------------------

5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

## Задание №8

### Ответить на вопросы:

**Вопрос 1** Для решения любой задачи с помощью компьютера необходимо выполнить следующие этапы:

*Установите правильную последовательность этапов.*

1	принятие решений;
2	программирование;
3	математическое моделирование;
4	алгоритмизация задач;
5	постановка задачи;
6	анализ результатов.

**Вопрос 2** Соотнесите свойства алгоритма с их описанием:

Результативность	- алгоритм должен приводить к решению задачи обязательно за конечное время;
Конечность	- неоднозначность толкования алгоритма недопустима;
Эффективность	- алгоритм должен обеспечить выдачу результата решения задачи на печать, на экран монитора или в файл;
Массовость	- правильный результат по алгоритму получен для одних исходных данных, то правильный результат по этому же алгоритму должен быть получен и для других исходных данных, допустимых в данной задаче;
Определенность	- позволяет решить задачу за приемлемое для разработчика время;

**Вопрос 3** Выберите тип алгоритма, описанного ниже:

Набор команд (указаний), выполняемых последовательно друг за другом

1. линейный;
2. разветвляющийся;
3. циклический.

**Вопрос 4** Выберите тип алгоритма, описанного ниже:

Алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения

1. линейный;
2. разветвляющийся;
3. циклический.

**Вопрос 5** Выберите тип алгоритма, описанного ниже:

Алгоритм, предусматривающий многократное повторение одного и того же действия над новыми исходными данными

1. линейный;
2. разветвляющийся;
3. циклический.

Оценка	Показатели оценки
3	Даны правильные ответы на 3 вопроса
4	Даны правильные ответы на 4 вопроса
5	Даны правильные ответы на 5 вопросов

### **Задание №9**

**Ответить на вопросы:**

1. Объясните на примере принцип работы полной и краткой формы условного оператора.
2. как показать условный оператор на схеме алгоритма? Приведите примеры.
3. Можно ли использовать условный оператор внутри другого условного оператора (например, в

ветви else)?

Оценка	Показатели оценки
3	Даны ответы на 2 вопроса, примеры не приведены
4	Даны ответы на все вопросы, примеры не приведены
5	Даны ответы на все вопросы, приведены примеры

### Задание №10

**Ответить на вопросы:**

1. Из чего состоит итерация для оператора while? Объясните на примере.
2. Объясните механизм работы оператора do-while на примерах.
3. Запишите в общем виде оператор цикла for. Какие ключевые слова при этом используются.

Оценка	Показатели оценки
3	Даны ответы на 2 вопроса, примеры не приведены
4	Даны ответы на все вопросы, примеры не приведены
5	Даны ответы на все вопросы, приведены примеры

### Задание №11

Ответить на тестовые вопросы, в личном кабинете ИАС "ИрКАТ" – " ОП.04 Основы алгоритмизации и программирования. Экзамен".

Оценка	Показатели оценки
5	Даны верные ответы на 95% вопросов.
4	Даны верные ответы на 75% вопросов.
3	Даны верные ответы на 50% вопросов.

### Задание №12

**Ответить на вопросы:**

**Вопрос 1** Установите правильную хронологию создания языков программирования:

1	C#;
2	бейсик (Basic);
3	паскаль (Pascal);

4	C++;
---	------

**Вопрос 2** Соотнесите годы создания языков программирования

1963	C#;
1971	бейсик (Basic);
1984	паскаль (Pascal);
2000	C++.

**Вопрос 3** Выберите процедурный язык программирования:

1. C++;
2. Basic;
3. Java.

**Вопрос 4** Какие языки программирования предназначены для решения задач искусственного интеллекта:

1. Commonlisp.
2. Planner.
3. Occam.
4. C++.
5. Java.

**Вопрос 5** Какие языки программирования предназначены для разработки программ-оболочек, разработки систем:

1. Commonlisp.
2. Planner.
3. Occam.
4. C++.
5. Java.

Оценка	Показатели оценки
3	Даны правильные ответы на 3 вопроса
4	Даны правильные ответы на 4 вопроса
5	Даны правильные ответы на 5 вопросов



### Задание №13

Вопрос 1:

Установите правильную хронологию создания языков программирования:

Ответ

1	Бейсик (Basic)
2	Паскаль (Pascal)
3	C++
4	C#

Вопрос 2:

*Соотнесите годы создания языков программирования*

Ответ

C#	2000
Бейсик (Basic)	1963
Паскаль (Pascal)	1971
C++	1986

Вопрос 3:

Выберите процедурный язык программирования:

Ответ

- C++
- Basic
- Java

Вопрос 4:

Какие языки программирования предназначены для решения задач искусственного интеллекта:

Ответ

- Commonlisp
- Planner
- Occam

- C++
- Java

Вопрос 5:

Какие языки программирования предназначены для разработки программ-оболочек, разработки систем:

Ответ

- Commonlisp
- Planner
- Оссам
- C++
- Java

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса
4	Даны верные ответы на 2 вопроса
3	Даны верные ответы на 1 вопрос

#### Задание №14

Вопрос 1:

Установите правильную хронологию создания языков программирования:

Ответ

1	Бейсик (Basic)
2	Паскаль (Pascal)
3	C++
4	C#

Вопрос 2:

*Соотнесите годы создания языков программирования*

Ответ

C#	2000
Бейсик (Basic)	1963
Паскаль (Pascal)	1971

Вопрос 3:

Выберите процедурный язык программирования:

Ответ

- C++
- Basic
- Java

Вопрос 4:

Какие языки программирования предназначены для решения задач искусственного интеллекта:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Вопрос 5:

Какие языки программирования предназначены для разработки программ-оболочек, разработки систем:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса
4	Даны верные ответы на 2 вопроса
3	Даны верные ответы на 1 вопрос

**Задание №15**

Вопрос 1:

Установите правильную хронологию создания языков программирования:

Ответ

1	Бейсик (Basic)
2	Паскаль (Pascal)
3	C++
4	C#

Вопрос 2:

*Соотнесите годы создания языков программирования*

Ответ

C#	2000
Бейсик (Basic)	1963
Паскаль (Pascal)	1971
C++	1986

Вопрос 3:

Выберите процедурный язык программирования:

Ответ

- C++
- Basic
- Java

Вопрос 4:

Какие языки программирования предназначены для решения задач искусственного интеллекта:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Вопрос 5:

Какие языки программирования предназначены для разработки программ-оболочек, разработки систем:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса
4	Даны верные ответы на 2 вопроса
3	Дан верный ответ на 1 вопрос.

### Задание №16

Вопрос 1:

Установите правильную хронологию создания языков программирования:

Ответ

1	Бейсик (Basic)
2	Паскаль (Pascal)
3	C++
4	C#

Вопрос 2:

*Соотнесите годы создания языков программирования*

Ответ

C#	2000
Бейсик (Basic)	1963
Паскаль (Pascal)	1971
C++	1986

Вопрос 3:

Выберите процедурный язык программирования:

Ответ

- C++
- Basic
- Java

Вопрос 4:

Какие языки программирования предназначены для решения задач искусственного интеллекта:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Вопрос 5:

Какие языки программирования предназначены для разработки программ-оболочек, разработки систем:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса
4	Даны верные ответы на 2 вопроса
3	Дан верный ответ на 1 вопрос.

### Задание №17

Вопрос 1:

Установите правильную хронологию создания языков программирования:

Ответ

1	Бейсик (Basic)
2	Паскаль (Pascal)
3	C++
4	C#

Вопрос 2:

*Соотнесите годы создания языков программирования*

Ответ

C#	2000
Бейсик (Basic)	1963
Паскаль (Pascal)	1971
C++	1986

Вопрос 3:

Выберите процедурный язык программирования:

Ответ

- C++
- Basic
- Java

Вопрос 4:

Какие языки программирования предназначены для решения задач искусственного интеллекта:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Вопрос 5:

Какие языки программирования предназначены для разработки программ-оболочек, разработки

систем:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса
4	Даны верные ответы на 2 вопроса
3	Дан верный ответ на 1 вопрос.

### Задание №18

Вопрос 1:

Установите правильную хронологию создания языков программирования:

Ответ

1	Бейсик (Basic)
2	Паскаль (Pascal)
3	C++
4	C#

Вопрос 2:

*Соотнесите годы создания языков программирования*

Ответ

C#	2000
Бейсик (Basic)	1963
Паскаль (Pascal)	1971
C++	1986

Вопрос 3:



Выберите процедурный язык программирования:

Ответ

- C++
- Basic
- Java

Вопрос 4:

Какие языки программирования предназначены для решения задач искусственного интеллекта:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Вопрос 5:

Какие языки программирования предназначены для разработки программ-оболочек, разработки систем:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса
4	Даны верные ответы на 2 вопроса
3	Дан верный ответ на 1 вопрос.

### Задание №19

Вопрос 1:

Установите правильную хронологию создания языков программирования:

Ответ

1	Бейсик (Basic)
2	Паскаль (Pascal)
3	C++
4	C#

Вопрос 2:

*Соотнесите годы создания языков программирования*

Ответ

C#	2000
Бейсик (Basic)	1963
Паскаль (Pascal)	1971
C++	1986

Вопрос 3:

Выберите процедурный язык программирования:

Ответ

- C++
- Basic
- Java

Вопрос 4:

Какие языки программирования предназначены для решения задач искусственного интеллекта:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Вопрос 5:

Какие языки программирования предназначены для разработки программ-оболочек, разработки систем:

Ответ

- Commonlisp
- Planner
- Occam
- C++
- Java

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса
4	Даны верные ответы на 2 вопроса
3	Дан верный ответ на 1 вопрос.

### Задание №20

Ответить на тестовые вопросы, в личном кабинете ИАС "ИркАТ" – " ОП.04 Основы алгоритмизации и программирования. Экзамен".

Оценка	Показатели оценки
5	Даны верные ответы на 95% вопросов.
4	Даны верные ответы на 75% вопросов.
3	Даны верные ответы на 50% вопросов.

### Задание №21

**Ответить на вопросы:**

**Вопрос 1.** Выберите видовые отличия термина "Язык программирования"

1. предназначен для записи программ;
2. имеет определенный синтаксис;
3. предназначен для передачи информации;
4. машиноориентированный, низкого уровня.

**Вопрос 2** Выберите видовые отличия термина "Алгоритм"

1. ориентирована на пользователя;
2. выполняется за конечное число шагов;
3. предназначен для записи программ;
4. имеет определенный синтаксис.

**Вопрос 3.** Выберите правильное определение для термина "Математическая модель"

1. модель, описывающая математическими формулами соотношения между количественными характеристиками объекта моделирования
2. объект, который воспроизводит наиболее характерные свойства реального процесса или устройства или концепции
3. модель, которая определяет уровни взаимодействия систем и их стандартные названия и функции

**Вопрос 4** Выберите родовое понятие для термина "Моделирование"

1. система познания;
2. набор объектов;
3. множество методов.

**Вопрос 5** Соотнесите термины и родовые понятия

алфавит		множество
идентификатор		набор символов
константа		величина
ключевые слова		единица языка

**Вопрос 6** Соотнесите термины и видовые отличия

идентификатор		является именем объектов программы
константа		имеет постоянное значение
ключевые слова		имеют специальное значение для компилятора

**Вопрос 7** Выберите правильный вариант использования условного оператора if для нахождения  $\text{MAX}\{C*D, E+F\}$ :

1. `if (C*D>E+F) Max=C*D; else Max:=E+F;`
2. `X=C*D; Y=E+F; if (X>Y) Max=X; else MAX=Y;`

3. оба варианта правильные.

**Вопрос 8** Выберите пример правильного идентификатора в языке C++

1. Fr\_5;
2. 10Sd;
3. scanf.

**Вопрос 9** Какие значения примут переменные S, N и P после выполнения следующего фрагмента программы,

если в переменную Y поочередно ввести следующие значения: -2; 0; -5; 6; 0?

...

```
S=0; N=0; P=1;
```

```
for (i=1; i<=5; i++)
```

```
{
```

```
scanf ("%d", &Y);
```

```
if (Y>0)
```

```
S=S+Y;
```

```
else if (Y=0)
```

```
N=N+1;
```

```
else P=P*Y;
```

```
}
```

...

1. S=-7; N=1; P=0;
2. S=6; N=2; P=10;
3. S=-1; N=4; P=10.

**Вопрос 10** Что делает следующий фрагмент программы с массивом A, содержащим N элементов?

```

...
for (i=0; i<N/2; i++)
{ M=A[i];
A[i]=A[N-1-i];
A[N-1-i]=M;
}
...

```

1. присваивает каждому элементу массива значение следующего элемента, а последнему - значение 1-го элемента;
2. записывает элементы массива в обратном порядке;
3. фрагмент содержит ошибки и работать не будет.

**Вопрос 11** Укажите правильный вариант определения класса в программе:

1. class Test  
{  
public:  
int a;  
};

2. class Test  
{  
private:  
float x;  
}

3. оба варианта правильные.

Оценка	Показатели оценки
3	Даны правильные ответы на 5-7 вопросов
4	Даны правильные ответы на 8-10 вопросов
5	Даны правильные ответы на 11 вопросов

## Задание №22

**Ответить на вопросы:**

1. Что такое зарезервированные (ключевые слова)? Приведите примеры ключевых слов.
2. Верно ли, что в любой программе на C++ обязательно должны использоваться переменные и

константы? Почему?

3. Чем отличаются операции "/" и "%"? Приведите примеры.

Оценка	Показатели оценки
3	Даны ответы на 2 вопроса, примеры не приведены
4	Даны ответы на все вопросы, примеры не приведены
5	Даны ответы на все вопросы, приведены примеры

### Задание №23

Вопрос 1:

Выберите правильный вариант использования условного оператора if для нахождения  $\text{MAX}\{C*D, E+F\}$ :

Ответ

- `f (C*D>E+F) Max=C*D; else Max:=E+F;`
- `X=C*D; Y=E+F; if (X>Y) Max=X; else MAX=Y;`
- оба варианта правильные

Вопрос 2:

Выберите пример правильного идентификатора в языке C++

Ответ

- `Fr_5`
- `10Sd`
- `scanf`

Вопрос 3:

Какие значения примут переменные S, N и P после выполнения следующего фрагмента программы, если в переменную Y поочередно ввести следующие значения: -2; 0; -5; 6; 0?

...

```
S=0; N=0; P=1;
```

```
for (i=1; i<=5; i++)
```

```
{
```

```
scanf ("%d", &Y);
```

```
if (Y>0)
```

```
S=S+Y;
```

```
else if (Y=0)
```

```
N=N+1;
```

```
else P=P*Y;
```

```
}
```

```
...
```

Ответ

- S=-7; N=1; P=0;
- S=6; N=2; P=10;
- S=-1; N=4; P=10;

Вопрос 4:

Что делает следующий фрагмент программы с массивом A, содержащим N элементов?

```
...
```

```
for (i=0; i<N/2; i++)
```

```
{ M=A[i];
```

```
A[i]=A[N-1-i];
```

```
A[N-1-i]=M;
```

```
}
```

```
...
```

Ответ

- присваивает каждому элементу массива значение следующего элемента, а последнему - значение 1-го элемента
- записывает элементы массива в обратном порядке
- фрагмент содержит ошибки и работать не будет

Вопрос 5:

Дан массив A=(-9 0 2 -6 5 4). Что будет сформировано в переменной X после выполнения



следующего фрагмента программы?

...

```
X=0;
```

```
for (i=0; i<6; i++)
```

```
if ((A[i]<0) && (A[i] %2 != 0))
```

```
X=X+A[i];
```

...

Ответ

- сумма отрицательных нечетных элементов целочисленного массива A ( $X = -9$ )
- произведение положительных четных элементов целочисленного массива A ( $X = 8$ )
- количество ненулевых элементов массива A ( $X = 5$ )

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Даны верные ответы на 1 вопрос.

#### Задание №24

Вопрос 1:

Выберите правильный вариант использования условного оператора if для нахождения  $\text{MAX}\{C*D, E+F\}$ :

Ответ

- `f (C*D>E+F) Max=C*D; else Max:=E+F;`
- `X=C*D; Y=E+F; if (X>Y) Max=X; else MAX=Y;`
- оба варианта правильные

Вопрос 2:

Выберите пример правильного идентификатора в языке C++

Ответ

- `Fr_5`
- `10Sd`

- scanf

Вопрос 3:

Какие значения примут переменные S, N и P после выполнения следующего фрагмента программы, если в переменную Y поочередно ввести следующие значения: -2; 0; -5; 6; 0?

...

```
S=0; N=0; P=1;
```

```
for (i=1; i<=5; i++)
```

```
{
```

```
scanf ("%d", &Y);
```

```
if (Y>0)
```

```
S=S+Y;
```

```
else if (Y=0)
```

```
N=N+1;
```

```
else P=P*Y;
```

```
}
```

...

Ответ

- S=-7; N=1; P=0;
- S=6; N=2; P=10;
- S=-1; N=4; P=10;

Вопрос 4:

Что делает следующий фрагмент программы с массивом A, содержащим N элементов?

...

```
for (i=0; i<N/2; i++)
```

```
{ M=A[i];
```

```
A[i]=A[N-1-i];
```

A[N-1-i]=M;

}

...

Ответ

- присваивает каждому элементу массива значение следующего элемента, а последнему - значение 1-го элемента
- записывает элементы массива в обратном порядке
- фрагмент содержит ошибки и работать не будет

Вопрос 5:

Дан массив A=(-9 0 2 -6 5 4). Что будет сформировано в переменной X после выполнения следующего фрагмента программы?

...

X=0;

for (i=0; i<6; i++)

if ((A[i]<0) && (A[i] %2 != 0))

X=X+A[i];

...

Ответ

- сумма отрицательных нечетных элементов целочисленного массива A (X = -9)
- произведение положительных четных элементов целочисленного массива A (X = 8)
- количество ненулевых элементов массива A (X = 5)

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верные ответы на 1 вопрос.

## Задание №25

Вопрос 1:

Выберите правильный вариант использования условного оператора if для нахождения  $\text{MAX}\{C*D,$

E+F}:

Ответ

- `f (C*D>E+F) Max=C*D; else Max:=E+F;`
- `X=C*D; Y=E+F; if (X>Y) Max=X; else MAX=Y;`
- оба варианта правильные

Вопрос 2:

Выберите пример правильного идентификатора в языке C++

Ответ

- `Fr_5`
- `10Sd`
- `scanf`

Вопрос 3:

Какие значения примут переменные S, N и P после выполнения следующего фрагмента программы, если в переменную Y поочередно ввести следующие значения: -2; 0; -5; 6; 0?

...

```
S=0; N=0; P=1;
```

```
for (i=1; i<=5; i++)
```

```
{
```

```
scanf ("%d", &Y);
```

```
if (Y>0)
```

```
S=S+Y;
```

```
else if (Y=0)
```

```
N=N+1;
```

```
else P=P*Y;
```

```
}
```

...

Ответ

- S=-7; N=1; P=0;
- S=6; N=2; P=10;
- S=-1; N=4; P=10;

Вопрос 4:

Что делает следующий фрагмент программы с массивом A, содержащим N элементов?

...

```
for (i=0; i<N/2; i++)
```

```
{ M=A[i];
```

```
A[i]=A[N-1-i];
```

```
A[N-1-i]=M;
```

```
}
```

...

Ответ

- присваивает каждому элементу массива значение следующего элемента, а последнему - значение 1-го элемента
- записывает элементы массива в обратном порядке
- фрагмент содержит ошибки и работать не будет

Вопрос 5:

Дан массив A=(-9 0 2 -6 5 4). Что будет сформировано в переменной X после выполнения следующего фрагмента программы?

...

```
X=0;
```

```
for (i=0; i<6; i++)
```

```
if ((A[i]<0) && (A[i] %2 != 0))
```

```
X=X+A[i];
```

...

Ответ

- сумма отрицательных нечетных элементов целочисленного массива А ( $X = -9$ )
- произведение положительных четных элементов целочисленного массива А ( $X = 8$ )
- количество ненулевых элементов массива А ( $X = 5$ )

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №26

Вопрос 1:

Выберите правильный вариант использования условного оператора if для нахождения  $\text{MAX}\{C*D, E+F\}$ :

Ответ

- `f (C*D>E+F) Max=C*D; else Max:=E+F;`
- `X=C*D; Y=E+F; if (X>Y) Max=X; else MAX=Y;`
- оба варианта правильные

Вопрос 2:

Выберите пример правильного идентификатора в языке C++

Ответ

- `Fr_5`
- `10Sd`
- `scanf`

Вопрос 3:

Какие значения примут переменные S, N и P после выполнения следующего фрагмента программы, если в переменную Y поочередно ввести следующие значения: -2; 0; -5; 6; 0?

...

`S=0; N=0; P=1;`

`for (i=1; i<=5; i++)`

`{`

`scanf ("%d", &Y);`

```
if (Y>0)
```

```
S=S+Y;
```

```
else if (Y=0)
```

```
N=N+1;
```

```
else P=P*Y;
```

```
}
```

```
...
```

Ответ

- S=-7; N=1; P=0;
- S=6; N=2; P=10;
- S=-1; N=4; P=10;

Вопрос 4:

Что делает следующий фрагмент программы с массивом A, содержащим N элементов?

```
...
```

```
for (i=0; i<N/2; i++)
```

```
{ M=A[i];
```

```
A[i]=A[N-1-i];
```

```
A[N-1-i]=M;
```

```
}
```

```
...
```

Ответ

- присваивает каждому элементу массива значение следующего элемента, а последнему - значение 1-го элемента
- записывает элементы массива в обратном порядке
- фрагмент содержит ошибки и работать не будет

Вопрос 5:

Дан массив A=(-9 0 2 -6 5 4). Что будет сформировано в переменной X после выполнения

следующего фрагмента программы?

...

```
X=0;
```

```
for (i=0; i<6; i++)
```

```
if ((A[i]<0) && (A[i] %2 != 0))
```

```
X=X+A[i];
```

...

Ответ

- сумма отрицательных нечетных элементов целочисленного массива A ( $X = -9$ )
- произведение положительных четных элементов целочисленного массива A ( $X = 8$ )
- количество ненулевых элементов массива A ( $X = 5$ )

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №27

Вопрос 1:

Выберите правильный вариант использования условного оператора if для нахождения  $\text{MAX}\{C*D, E+F\}$ :

Ответ

- `f (C*D>E+F) Max=C*D; else Max:=E+F;`
- `X=C*D; Y=E+F; if (X>Y) Max=X; else MAX=Y;`
- оба варианта правильные

Вопрос 2:

Выберите пример правильного идентификатора в языке C++

Ответ

- `Fr_5`
- `10Sd`



- scanf

Вопрос 3:

Какие значения примут переменные S, N и P после выполнения следующего фрагмента программы, если в переменную Y поочередно ввести следующие значения: -2; 0; -5; 6; 0?

...

```
S=0; N=0; P=1;
```

```
for (i=1; i<=5; i++)
```

```
{
```

```
scanf ("%d", &Y);
```

```
if (Y>0)
```

```
S=S+Y;
```

```
else if (Y=0)
```

```
N=N+1;
```

```
else P=P*Y;
```

```
}
```

...

Ответ

- S=-7; N=1; P=0;
- S=6; N=2; P=10;
- S=-1; N=4; P=10;

Вопрос 4:

Что делает следующий фрагмент программы с массивом A, содержащим N элементов?

...

```
for (i=0; i<N/2; i++)
```

```
{ M=A[i];
```

```
A[i]=A[N-1-i];
```

A[N-1-i]=M;

}

...

Ответ

- присваивает каждому элементу массива значение следующего элемента, а последнему - значение 1-го элемента
- записывает элементы массива в обратном порядке
- фрагмент содержит ошибки и работать не будет

Вопрос 5:

Дан массив A=(-9 0 2 -6 5 4). Что будет сформировано в переменной X после выполнения следующего фрагмента программы?

...

X=0;

for (i=0; i<6; i++)

if ((A[i]<0) && (A[i] %2 != 0))

X=X+A[i];

...

Ответ

- сумма отрицательных нечетных элементов целочисленного массива A (X = -9)
- произведение положительных четных элементов целочисленного массива A (X = 8)
- количество ненулевых элементов массива A (X = 5)

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

## Задание №28

Вопрос 1:

Выберите правильный вариант использования условного оператора if для нахождения  $\text{MAX}\{C*D,$

E+F}:

Ответ

- $f(C*D > E+F) \text{ Max} = C*D; \text{ else Max} := E+F;$
- $X = C*D; Y = E+F; \text{ if } (X > Y) \text{ Max} = X; \text{ else MAX} = Y;$
- оба варианта правильные

Вопрос 2:

Выберите пример правильного идентификатора в языке C++

Ответ

- Fr\_5
- 10Sd
- scanf

Вопрос 3:

Какие значения примут переменные S, N и P после выполнения следующего фрагмента программы, если в переменную Y поочередно ввести следующие значения: -2; 0; -5; 6; 0?

...

```
S=0; N=0; P=1;
```

```
for (i=1; i<=5; i++)
```

```
{
```

```
scanf ("%d", &Y);
```

```
if (Y>0)
```

```
S=S+Y;
```

```
else if (Y=0)
```

```
N=N+1;
```

```
else P=P*Y;
```

```
}
```

...

Ответ

- S=-7; N=1; P=0;
- S=6; N=2; P=10;
- S=-1; N=4; P=10;

Вопрос 4:

Что делает следующий фрагмент программы с массивом A, содержащим N элементов?

...

```
for (i=0; i<N/2; i++)
```

```
{ M=A[i];
```

```
A[i]=A[N-1-i];
```

```
A[N-1-i]=M;
```

```
}
```

...

Ответ

- присваивает каждому элементу массива значение следующего элемента, а последнему - значение 1-го элемента
- записывает элементы массива в обратном порядке
- фрагмент содержит ошибки и работать не будет

Вопрос 5:

Дан массив A=(-9 0 2 -6 5 4). Что будет сформировано в переменной X после выполнения следующего фрагмента программы?

...

```
X=0;
```

```
for (i=0; i<6; i++)
```

```
if ((A[i]<0) && (A[i] %2 != 0))
```

```
X=X+A[i];
```

...

Ответ

- сумма отрицательных нечетных элементов целочисленного массива А ( $X = -9$ )
- произведение положительных четных элементов целочисленного массива А ( $X = 8$ )
- количество ненулевых элементов массива А ( $X = 5$ )

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №29

Вопрос 1:

Выберите правильный вариант использования условного оператора if для нахождения  $\text{MAX}\{C*D, E+F\}$ :

Ответ

- `f (C*D>E+F) Max=C*D; else Max:=E+F;`
- `X=C*D; Y=E+F; if (X>Y) Max=X; else MAX=Y;`
- оба варианта правильные

Вопрос 2:

Выберите пример правильного идентификатора в языке C++

Ответ

- `Fr_5`
- `10Sd`
- `scanf`

Вопрос 3:

Какие значения примут переменные S, N и P после выполнения следующего фрагмента программы, если в переменную Y поочередно ввести следующие значения: -2; 0; -5; 6; 0?

...

`S=0; N=0; P=1;`

`for (i=1; i<=5; i++)`

`{`

`scanf ("%d", &Y);`

```
if (Y>0)
```

```
S=S+Y;
```

```
else if (Y=0)
```

```
N=N+1;
```

```
else P=P*Y;
```

```
}
```

```
...
```

Ответ

- S=-7; N=1; P=0;
- S=6; N=2; P=10;
- S=-1; N=4; P=10;

Вопрос 4:

Что делает следующий фрагмент программы с массивом A, содержащим N элементов?

```
...
```

```
for (i=0; i<N/2; i++)
```

```
{ M=A[i];
```

```
A[i]=A[N-1-i];
```

```
A[N-1-i]=M;
```

```
}
```

```
...
```

Ответ

- присваивает каждому элементу массива значение следующего элемента, а последнему - значение 1-го элемента
- записывает элементы массива в обратном порядке
- фрагмент содержит ошибки и работать не будет

Вопрос 5:

Дан массив A=(-9 0 2 -6 5 4). Что будет сформировано в переменной X после выполнения

следующего фрагмента программы?

...

X=0;

for (i=0; i<6; i++)

if ((A[i]<0) && (A[i] %2 != 0))

X=X+A[i];

...

Ответ

- сумма отрицательных нечетных элементов целочисленного массива A (X = -9)
- произведение положительных четных элементов целочисленного массива A (X = 8)
- количество ненулевых элементов массива A (X = 5)

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №30

**Ответить на вопросы:**

1. Данные каких типов могут быть элементами массива? Приведите примеры.
2. Как получить доступ к конкретному элементу массива? Приведите примеры.
3. Можно ли в программе объявить пользовательский тип "массива"? Как это сделать? Приведите примеры.

Оценка	Показатели оценки
3	Даны ответы на 2 вопроса, примеры не приведены
4	Даны ответы на все вопросы, примеры не приведены
5	Даны ответы на все вопросы, приведены примеры

### Задание №31

Ответить на тестовые вопросы, в личном кабинете ИАС "ИркАТ" – " ОП.04 Основы

алгоритмизации и программирования. Экзамен".

Оценка	Показатели оценки
5	Даны верные ответы на 95% вопросов.
4	Даны верные ответы на 75% вопросов.
3	Даны верные ответы на 50% вопросов.

### Задание №32

#### Ответить на вопросы:

**Вопрос 1.** В чем разница между формальными и фактическими параметрами?

1. никакой разницы нет;
2. формальные параметры используются при описании подпрограммы, а фактические — при вызове подпрограммы;
3. фактические параметры используются при описании подпрограммы, а формальные — при вызове подпрограммы.

**Вопрос 2.** В чем разница между глобальными и локальными переменными?

1. разницы нет;
2. глобальные переменные не могут использоваться в подпрограммах, для этого служат локальные переменные;
3. глобальные переменные могут использоваться во всех подпрограммах и в функции `main ()`, а локальные переменные только в своей подпрограмме.

**Вопрос 3.** Выберите определение функции, не содержащее ошибок:

1. `int Area(int A, int B)`  
`{ float S;`  
`Area=A*B;}`

2. `int Area(int A, int B)`  
`{ int S;`  
`S=A*B; return S;}`

3. `int Area(int A, int B)`  
`{ int S;`  
`S=A*B;`  
`return Area;`



**Вопрос 4.** Что будет выведено на экран в результате работы следующей программы?

```
int A;  
  
void Prim(int A)  
{A=5;  
printf(" %d",A);  
}  
  
int main()  
(int A=10;  
Prim(A);  
printf ( " %d", A) ; return 0;  
}
```

1. 10 10;
2. 5 10;
3. 5 5.

**Вопрос 5.** Должны ли имена параметров, указанных в прототипе, определении и вызове функции, соответствовать друг другу?

1. да;
2. нет;
3. должны соответствовать в прототипе и определении.

Оценка	Показатели оценки
3	Даны правильные ответы на 3 вопроса
4	Даны правильные ответы на 4 вопроса
5	Даны правильные ответы на 5 вопросов

### Задание №33

**Ответить на вопросы:**

1. Что такое возвращаемое значение в функции? Приведите примеры.
2. Напишите структуру функции в общем виде и объясните, для чего используется каждый раздел функции. Приведите примеры определения функции.
3. Что такое формальные и фактические параметры в функции? Приведите примеры.

Оценка	Показатели оценки
3	Даны ответы на 2 вопроса, примеры не приведены
4	Даны ответы на все вопросы, примеры не приведены
5	Даны ответы на все вопросы, приведены примеры

#### Задание №34

##### Ответить на вопросы:

1. Что такое "структура" в программировании с точки зрения типов данных?
2. Какие требования предъявляются к именам полей структур? Поясните на примерах.
3. Как правильно присваивать значения полям структуры? Поясните на примерах.

Оценка	Показатели оценки
3	Даны ответы на 2 вопроса, примеры не приведены
4	Даны ответы на все вопросы, примеры не приведены
5	Даны ответы на все вопросы, приведены примеры

#### Задание №35

Вопрос 1:

В чем разница между формальными и фактическими параметрами?

Ответ

- фактические параметры используются при описании подпрограммы, а формальные — при вызове подпрограммы.
- никакой разницы нет;
- формальные параметры используются при описании подпрограммы, а фактические — при вызове подпрограммы;

Вопрос 2:

В чем разница между глобальными и локальными переменными?

Ответ

- разницы нет;
- глобальные переменные не могут использоваться в подпрограммах, для этого служат локальные переменные;
- глобальные переменные могут использоваться во всех подпрограммах и в функции main (), а локальные переменные только в своей подпрограмме.

Вопрос 3:

Выберите определение функции, не содержащее ошибок:

Ответ

- int Area(int A, int B) { float S; Area=A\*B;}
- int Area(int A, int B) { int S; S=A\*B; return S;}
- int Area(int A, int B) { int S; S=A\*B; return Area;

Вопрос 4:

*Что будет выведено на экран в результате работы следующей программы?*

```
int A;
```

```
void Prim(int A)
```

```
{A=5;
```

```
printf(" %d",A);
```

```
}
```

```
int main()
```

```
(int A=10;
```

```
Prim(A);
```

```
printf ( " %d", A) ; return 0;
```

```
}
```

Ответ

- 10 10
- 5 10
- 5 5

Вопрос 5:

Должны ли имена параметров, указанных в прототипе, определении и вызове функции, соответствовать друг другу?

Ответ

- да
- нет
- должны соответствовать в прототипе и определении

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Даны верные ответы на 1 вопрос.

### Задание №36

Вопрос 1:

В чем разница между формальными и фактическими параметрами?

Ответ

- фактические параметры используются при описании подпрограммы, а формальные — при вызове подпрограммы.
- никакой разницы нет;
- формальные параметры используются при описании подпрограммы, а фактические — при вызове подпрограммы;

Вопрос 2:

В чем разница между глобальными и локальными переменными?

Ответ

- разницы нет;
- глобальные переменные не могут использоваться в подпрограммах, для этого служат локальные переменные;
- глобальные переменные могут использоваться во всех подпрограммах и в функции `main()`, а локальные переменные только в своей подпрограмме.

Вопрос 3:

Выберите определение функции, не содержащее ошибок:

Ответ

- `int Area(int A, int B) { float S; Area=A*B;}`
- `int Area(int A, int B) { int S; S=A*B; return S;}`
- `int Area(int A, int B) { int S; S=A*B; return Area;`

Вопрос 4:

*Что будет выведено на экран в результате работы следующей программы?*

```
int A;  
  
void Prim(int A)  
{A=5;  
  
printf(" %d",A);  
  
}  
  
int main()  
(int A=10;  
  
Prim(A);  
  
printf ( " %d", A) ; return 0;  
  
}
```

Ответ

- 10 10
- 5 10
- 5 5

Вопрос 5:

Должны ли имена параметров, указанных в прототипе, определении и вызове функции, соответствовать друг другу?

Ответ

- да
- нет
- должны соответствовать в прототипе и определении

Оценка	Показатели оценки

5	Даны верные ответы на 3 вопроса
4	Даны верные ответы на 2 вопроса
3	Дан верный ответ на 1 вопрос

### Задание №37

Вопрос 1:

В чем разница между формальными и фактическими параметрами?

Ответ

- фактические параметры используются при описании подпрограммы, а формальные — при вызове подпрограммы.
- никакой разницы нет;
- формальные параметры используются при описании подпрограммы, а фактические — при вызове подпрограммы;

Вопрос 2:

В чем разница между глобальными и локальными переменными?

Ответ

- разницы нет;
- глобальные переменные не могут использоваться в подпрограммах, для этого служат локальные переменные;
- глобальные переменные могут использоваться во всех подпрограммах и в функции `main()`, а локальные переменные только в своей подпрограмме.

Вопрос 3:

Выберите определение функции, не содержащее ошибок:

Ответ

- `int Area(int A, int B) { float S; Area=A*B;}`
- `int Area(int A, int B) { int S; S=A*B; return S;}`
- `int Area(int A, int B) { int S; S=A*B; return Area;`

Вопрос 4:

*Что будет выведено на экран в результате работы следующей программы?*

`int A;`

```

void Prim(int A)

{A=5;

printf(" %d",A);

}

int main()

(int A=10;

Prim(A);

printf ( " %d", A) ; return 0;

}

```

Ответ

- 10 10
- 5 10
- 5 5

Вопрос 5:

Должны ли имена параметров, указанных в прототипе, определении и вызове функции, соответствовать друг другу?

Ответ

- да
- нет
- должны соответствовать в прототипе и определении

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №38

Вопрос 1:

В чем разница между формальными и фактическими параметрами?

Ответ

- фактические параметры используются при описании подпрограммы, а формальные — при вызове подпрограммы.
- никакой разницы нет;
- формальные параметры используются при описании подпрограммы, а фактические — при вызове подпрограммы;

Вопрос 2:

В чем разница между глобальными и локальными переменными?

Ответ

- разницы нет;
- глобальные переменные не могут использоваться в подпрограммах, для этого служат локальные переменные;
- глобальные переменные могут использоваться во всех подпрограммах и в функции `main ()`, а локальные переменные только в своей подпрограмме.

Вопрос 3:

Выберите определение функции, не содержащее ошибок:

Ответ

- `int Area(int A, int B) { float S; Area=A*B;}`
- `int Area(int A, int B) { int S; S=A*B; return S;}`
- `int Area(int A, int B) { int S; S=A*B; return Area;`

Вопрос 4:

*Что будет выведено на экран в результате работы следующей программы?*

```
int A;
```

```
void Prim(int A)
```

```
{A=5;
```

```
printf(" %d",A);
```

```
}
```

```
int main()
```

```
(int A=10;
```



```

Prim(A);

printf ( " %d", A) ; return 0;

}

```

Ответ

- 10 10
- 5 10
- 5 5

Вопрос 5:

Должны ли имена параметров, указанных в прототипе, определении и вызове функции, соответствовать друг другу?

Ответ

- да
- нет
- должны соответствовать в прототипе и определении

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №39

Вопрос 1:

В чем разница между формальными и фактическими параметрами?

Ответ

- фактические параметры используются при описании подпрограммы, а формальные — при вызове подпрограммы.
- никакой разницы нет;
- формальные параметры используются при описании подпрограммы, а фактические — при вызове подпрограммы;

Вопрос 2:

В чем разница между глобальными и локальными переменными?

Ответ

- разницы нет;
- глобальные переменные не могут использоваться в подпрограммах, для этого служат локальные переменные;
- глобальные переменные могут использоваться во всех подпрограммах и в функции `main ()`, а локальные переменные только в своей подпрограмме.

Вопрос 3:

Выберите определение функции, не содержащее ошибок:

Ответ

- `int Area(int A, int B) { float S; Area=A*B;}`
- `int Area(int A, int B) { int S; S=A*B; return S;}`
- `int Area(int A, int B) { int S; S=A*B; return Area;`

Вопрос 4:

*Что будет выведено на экран в результате работы следующей программы?*

```
int A;
```

```
void Prim(int A)
```

```
{A=5;
```

```
printf(" %d",A);
```

```
}
```

```
int main()
```

```
(int A=10;
```

```
Prim(A);
```

```
printf ( " %d", A) ; return 0;
```

```
}
```

Ответ

- 10 10
- 5 10
- 5 5

Вопрос 5:

Должны ли имена параметров, указанных в прототипе, определении и вызове функции, соответствовать друг другу?

Ответ

- да
- нет
- должны соответствовать в прототипе и определении

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

#### Задание №40

Вопрос 1:

В чем разница между формальными и фактическими параметрами?

Ответ

- фактические параметры используются при описании подпрограммы, а формальные — при вызове подпрограммы.
- никакой разницы нет;
- формальные параметры используются при описании подпрограммы, а фактические — при вызове подпрограммы;

Вопрос 2:

В чем разница между глобальными и локальными переменными?

Ответ

- разницы нет;
- глобальные переменные не могут использоваться в подпрограммах, для этого служат локальные переменные;
- глобальные переменные могут использоваться во всех подпрограммах и в функции `main()`, а локальные переменные только в своей подпрограмме.

Вопрос 3:

Выберите определение функции, не содержащее ошибок:

Ответ

- `int Area(int A, int B) { float S; Area=A*B;}`
- `int Area(int A, int B) { int S; S=A*B; return S;}`
- `int Area(int A, int B) { int S; S=A*B; return Area;`

Вопрос 4:

*Что будет выведено на экран в результате работы следующей программы?*

```
int A;  
  
void Prim(int A)  
{A=5;  
  
printf(" %d",A);  
  
}  
  
int main()  
(int A=10;  
  
Prim(A);  
  
printf ( " %d", A) ; return 0;  
  
}
```

Ответ

- 10 10
- 5 10
- 5 5

Вопрос 5:

Должны ли имена параметров, указанных в прототипе, определении и вызове функции, соответствовать друг другу?

Ответ

- да
- нет
- должны соответствовать в прототипе и определении

Оценка	Показатели оценки

5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

#### Задание №41

Вопрос 1:

В чем разница между формальными и фактическими параметрами?

Ответ

- фактические параметры используются при описании подпрограммы, а формальные — при вызове подпрограммы.
- никакой разницы нет;
- формальные параметры используются при описании подпрограммы, а фактические — при вызове подпрограммы;

Вопрос 2:

В чем разница между глобальными и локальными переменными?

Ответ

- разницы нет;
- глобальные переменные не могут использоваться в подпрограммах, для этого служат локальные переменные;
- глобальные переменные могут использоваться во всех подпрограммах и в функции main (), а локальные переменные только в своей подпрограмме.

Вопрос 3:

Выберите определение функции, не содержащее ошибок:

Ответ

- `int Area(int A, int B) { float S; Area=A*B;}`
- `int Area(int A, int B) { int S; S=A*B; return S;}`
- `int Area(int A, int B) { int S; S=A*B; return Area;`

Вопрос 4:

*Что будет выведено на экран в результате работы следующей программы?*

`int A;`

```

void Prim(int A)

{A=5;

printf(" %d",A);

}

int main()

(int A=10;

Prim(A);

printf ( " %d", A) ; return 0;

}

```

Ответ

- 10 10
- 5 10
- 5 5

Вопрос 5:

Должны ли имена параметров, указанных в прототипе, определении и вызове функции, соответствовать друг другу?

Ответ

- да
- нет
- должны соответствовать в прототипе и определении

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

## Задание №42

Ответить на тестовые вопросы, в личном кабинете ИАС "ИркАТ" – " ОП.04 Основы алгоритмизации и программирования. Экзамен".

Оценка	Показатели оценки

5	Даны верные ответы на 95% вопросов.
4	Даны верные ответы на 75% вопросов.
3	Даны верные ответы на 50% вопросов.

### Задание №43

#### Ответить на вопросы:

**Вопрос 1.** Принцип инкапсуляции обеспечивает:

1. объединение данных и методов работы с ними в классе;
2. доступ к членам класса;
3. сокрытие данных внутри класса.

**Вопрос 2.** Укажите правильный вариант инициализации целочисленных констант A и B в классе:

1. `class::klass(int a, int b) : A(a), B(b)`  
`{`  
`...`  
`}`

2. `class::klass(int a, int b) : A= a, B = b`  
`{`  
`...`  
`}`

3. `class::klass(int a, int b) : A(a) B(b)`  
`{`  
`...`  
`}`

**Вопрос 3.** Наследование — это:

1. возможность использования базовых библиотек языка C++ в своих программах;
2. условия, описывающие последовательность вызова конструкторов для объектов классов, используемых в программе;
3. механизм создания производных классов, на базе уже имеющихся.

**Вопрос 4.** Укажите верную последовательность выполнения деструкторов:

1. сначала выполняются операторы деструктора базового класса, затем операторы деструктора порожденного класса;
2. сначала выполняются операторы деструктора порожденного класса, затем операторы деструктора базового класса;
3. операторы деструктора порожденного класса выполняются одновременно с операторами деструктора базового класса.

**Вопрос 5.** Полиморфизм — это:

1. возможность программного кода работать с разными объектами одинаковым образом;
2. возможность изменения программного кода в зависимости от решаемых задач;
3. возможность доработки программного кода в случае необходимости.

Оценка	Показатели оценки
3	Даны правильные ответы на 3 вопроса
4	Даны правильные ответы на 4 вопроса
5	Даны правильные ответы на 5 вопросов

#### **Задание №44**

Вопрос 1:

Принцип инкапсуляции обеспечивает:

Ответ

- объединение данных и методов работы с ними в классе
- доступ к членам класса
- сокрытие данных внутри класса

Вопрос 2:

Что означает аббревиатура ООП?

Ответ

- объектный образ в программировании
- объектно-ориентированное программирование
- объективно ориентированное программирование

Вопрос 3:



Наследование — это:

Ответ

- возможность использования базовых библиотек языка C++ в своих программах
- условия, описывающие последовательность вызова конструкторов для объектов классов, используемых в программе
- механизм создания производных классов, на базе уже имеющихся

Вопрос 4:

Укажите верную последовательность выполнения деструкторов:

Ответ

- |   |                                             |
|---|---------------------------------------------|
| 1 | выполняет действия, описанные программистом |
| 2 | присваивает указателям пустые значения      |
| 3 | выполняет действия по разрушению объекта    |

Вопрос 5:

Полиморфизм — это

Ответ

- возможность программного кода работать с разными объектами одинаковым образом
- возможность изменения программного кода в зависимости от решаемых задач
- возможность доработки программного кода в случае необходимости

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Даны верные ответы на 1 вопрос.

### Задание №45

Вопрос 1:

Принцип инкапсуляции обеспечивает:

Ответ

- объединение данных и методов работы с ними в классе
- доступ к членам класса
- сокрытие данных внутри класса

Вопрос 2:

Что означает аббревиатура ООП?

Ответ

- объектный образ в программировании
- объектно-ориентированное программирование
- объективно ориентированное программирование

Вопрос 3:

Наследование — это:

Ответ

- возможность использования базовых библиотек языка C++ в своих программах
- условия, описывающие последовательность вызова конструкторов для объектов классов, используемых в программе
- механизм создания производных классов, на базе уже имеющихся

Вопрос 4:

Укажите верную последовательность выполнения деструкторов:

Ответ

- |   |                                             |
|---|---------------------------------------------|
| 1 | выполняет действия, описанные программистом |
| 2 | присваивает указателям пустые значения      |
| 3 | выполняет действия по разрушению объекта    |

Вопрос 5:

Полиморфизм — это

Ответ

- возможность программного кода работать с разными объектами одинаковым образом
- возможность изменения программного кода в зависимости от решаемых задач

- возможность доработки программного кода в случае необходимости

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

#### Задание №46

Вопрос 1:

Принцип инкапсуляции обеспечивает:

Ответ

- объединение данных и методов работы с ними в классе
- доступ к членам класса
- сокрытие данных внутри класса

Вопрос 2:

Что означает аббревиатура ООП?

Ответ

- объектный образ в программировании
- объектно-ориентированное программирование
- объективно ориентированное программирование

Вопрос 3:

Наследование — это:

Ответ

- возможность использования базовых библиотек языка C++ в своих программах
- условия, описывающие последовательность вызова конструкторов для объектов классов, используемых в программе
- механизм создания производных классов, на базе уже имеющихся

Вопрос 4:

Укажите верную последовательность выполнения деструкторов:

Ответ

1	выполняет действия, описанные программистом
2	присваивает указателям пустые значения
3	выполняет действия по разрушению объекта

Вопрос 5:

Полиморфизм — это

Ответ

- возможность программного кода работать с разными объектами одинаковым образом
- возможность изменения программного кода в зависимости от решаемых задач
- возможность доработки программного кода в случае необходимости

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Даны верные ответы на 1 вопрос.

#### Задание №47

Вопрос 1:

Принцип инкапсуляции обеспечивает:

Ответ

- объединение данных и методов работы с ними в классе
- доступ к членам класса
- сокрытие данных внутри класса

Вопрос 2:

Что означает аббревиатура ООП?

Ответ

- объектный образ в программировании
- объектно-ориентированное программирование
- объективно ориентированное программирование

Вопрос 3:

Наследование — это:

Ответ

- возможность использования базовых библиотек языка C++ в своих программах
- условия, описывающие последовательность вызова конструкторов для объектов классов, используемых в программе
- механизм создания производных классов, на базе уже имеющихся

Вопрос 4:

Укажите верную последовательность выполнения деструкторов:

Ответ

- |   |                                             |
|---|---------------------------------------------|
| 1 | выполняет действия, описанные программистом |
| 2 | присваивает указателям пустые значения      |
| 3 | выполняет действия по разрушению объекта    |

Вопрос 5:

Полиморфизм — это

Ответ

- возможность программного кода работать с разными объектами одинаковым образом
- возможность изменения программного кода в зависимости от решаемых задач
- возможность доработки программного кода в случае необходимости

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Даны верные ответы на 1 вопрос.

### Задание №48

Вопрос 1:

Принцип инкапсуляции обеспечивает:

Ответ

- объединение данных и методов работы с ними в классе
- доступ к членам класса
- сокрытие данных внутри класса

Вопрос 2:

Что означает аббревиатура ООП?

Ответ

- объектный образ в программировании
- объектно-ориентированное программирование
- объективно ориентированное программирование

Вопрос 3:

Наследование — это:

Ответ

- возможность использования базовых библиотек языка C++ в своих программах
- условия, описывающие последовательность вызова конструкторов для объектов классов, используемых в программе
- механизм создания производных классов, на базе уже имеющихся

Вопрос 4:

Укажите верную последовательность выполнения деструкторов:

Ответ

- |   |                                             |
|---|---------------------------------------------|
| 1 | выполняет действия, описанные программистом |
| 2 | присваивает указателям пустые значения      |
| 3 | выполняет действия по разрушению объекта    |

Вопрос 5:

Полиморфизм — это

Ответ

- возможность программного кода работать с разными объектами одинаковым образом
- возможность изменения программного кода в зависимости от решаемых задач
- возможность доработки программного кода в случае необходимости

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Даны верные ответы на 1 вопрос.

#### Задание №49

Вопрос 1:

Принцип инкапсуляции обеспечивает:

Ответ

- объединение данных и методов работы с ними в классе
- доступ к членам класса
- сокрытие данных внутри класса

Вопрос 2:

Что означает аббревиатура ООП?

Ответ

- объектный образ в программировании
- объектно-ориентированное программирование
- объективно ориентированное программирование

Вопрос 3:

Наследование — это:

Ответ

- возможность использования базовых библиотек языка C++ в своих программах
- условия, описывающие последовательность вызова конструкторов для объектов классов, используемых в программе
- механизм создания производных классов, на базе уже имеющихся

Вопрос 4:

Укажите верную последовательность выполнения деструкторов:

Ответ

- |   |                                             |
|---|---------------------------------------------|
| 1 | выполняет действия, описанные программистом |
| 2 | присваивает указателям пустые значения      |
| 3 | выполняет действия по разрушению объекта    |

Вопрос 5:

Полиморфизм — это

Ответ

- возможность программного кода работать с разными объектами одинаковым образом
- возможность изменения программного кода в зависимости от решаемых задач
- возможность доработки программного кода в случае необходимости

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

### Задание №50

Вопрос 1:

Принцип инкапсуляции обеспечивает:

Ответ

- объединение данных и методов работы с ними в классе
- доступ к членам класса
- сокрытие данных внутри класса

Вопрос 2:

Что означает аббревиатура ООП?

Ответ

- объектный образ в программировании
- объектно-ориентированное программирование



- объективно ориентированное программирование

Вопрос 3:

Наследование — это:

Ответ

- возможность использования базовых библиотек языка C++ в своих программах
- условия, описывающие последовательность вызова конструкторов для объектов классов, используемых в программе
- механизм создания производных классов, на базе уже имеющихся

Вопрос 4:

Укажите верную последовательность выполнения деструкторов:

Ответ

- |   |                                             |
|---|---------------------------------------------|
| 1 | выполняет действия, описанные программистом |
| 2 | присваивает указателям пустые значения      |
| 3 | выполняет действия по разрушению объекта    |

Вопрос 5:

Полиморфизм — это

Ответ

- возможность программного кода работать с разными объектами одинаковым образом
- возможность изменения программного кода в зависимости от решаемых задач
- возможность доработки программного кода в случае необходимости

Оценка	Показатели оценки
5	Даны верные ответы на 3 вопроса.
4	Даны верные ответы на 2 вопроса.
3	Дан верный ответ на 1 вопрос.

**Перечень практических заданий:**

**Задание №1**

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Напишите программу, массив в котором заполняется случайными числами в интервале  $[0,5]$ , вводится число  $X$  и находятся все значения, равные  $X$ .

Пример:

Массив:

13 15 13 34 5

Что ищем:

13

Нашли:  $A[1]=13$ ,  $A[3]=13$

Пример:

Массив:

5 7 9 34 6

Что ищем:

15

Ничего не нашли.

Оценка	Показатели оценки
5	Алгоритм построен верно.
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков).
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

## Задание №2

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Дано целое число  $N (> 0)$ . С помощью операций деления нацело и взятия остатка от деления определить, имеется ли в записи числа  $N$  цифра «2». Если имеется, то вывести TRUE, если нет — вывести FALSE.

Оценка	Показатели оценки
--------	-------------------

5	Алгоритм построен верно;
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков);
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №3

Разработайте алгоритм для задачи:

*Определить, принадлежит ли точка  $A\{x, y\}$  заданной фигуре.*

Оценка	Показатели оценки
3	Алгоритм составлен с двумя ошибками
4	Алгоритм составлен с одной ошибкой
5	Алгоритм составлен без ошибок

### Задание №4

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Программа, которая запрашивает стоимость заказа. Если стоимость заказа 1000 и более, то программа должна вывести “Доставка бесплатно.” Иначе программа должна спросить “1. Самовывоз. 2. Доставка курьером.”. Если пользователь вводит 1- программа должна вывести “Выбран самовывоз”, если 2 “Выбрана доставка курьером”. В конце программа должна вывести сумму заказа с учетом доставки. Доставка курьером увеличивает стоимость заказа на 300 руб., самовывоз увеличивает стоимость заказа на 100 руб. Программа должна вывести стоимость заказа (только цифру).

Оценка	Показатели оценки
5	Алгоритм построен верно;
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков);
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №5

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Некоторое предприятие ежедневно расходует  $X$  Квт/час электроэнергии – зимой,  $Y$  Квт/час – летом и  $Z$  квт/час - весной и осенью. Составить программу, вычисляющую расход электроэнергии  $R$  для заданного месяца текущего года.

Оценка	Показатели оценки
5	Алгоритм построен верно;
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков);
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №6

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Дано натуральное число.

а) Верно ли, что произведение его цифр больше  $b$ ?

б) Верно ли, что это число  $k$ -значное?

Составное условие и вложенный условный оператор не использовать

Оценка	Показатели оценки
5	Алгоритм построен верно;
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков);
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №7

Разработайте алгоритм для задачи:

*Ввести два числа и символ — знак арифметической операции. В зависимости от введенного знака операции вычислить значение арифметического выражения.*

Оценка	Показатели оценки

3	Алгоритм составлен с двумя ошибками
4	Алгоритм составлен с одной ошибкой
5	Алгоритм составлен без ошибок

### Задание №8

Разработайте алгоритм для задачи:

*Даны три числа:  $a$ ,  $b$ ,  $c$ . Определить, могут ли они быть сторонами треугольника, и если могут, то определить его тип: равносторонний, равнобедренный, разносторонний. (Условие существования треугольника: сумма длин любых двух сторон треугольника превышает длину 3-й стороны. Следует также учесть случай, когда длина одной из сторон равна нулю или имеет отрицательное значение.)*

Оценка	Показатели оценки
3	Алгоритм составлен с двумя ошибками
4	Алгоритм составлен с одной ошибкой
5	Алгоритм составлен без ошибок

### Задание №9

Разработайте алгоритм для задачи:

*Ввести два целых числа. Вывести в порядке убывания все числа, лежащие между ними, и количество этих чисел. Каждое третье число не печатать и не учитывать.*

Оценка	Показатели оценки
3	Алгоритм составлен с двумя ошибками
4	Алгоритм составлен с одной ошибкой
5	Алгоритм составлен без ошибок

### Задание №10

Разработайте алгоритм для задачи:

В 1202 г. итальянский математик Леонард Фибоначчи подсчитывал, на сколько увеличивается число кроликов в хозяйстве каждый год. При этом он получил последовательность такого вида: 1, 1, 2, 3, 5, 8, 13, 21, 34 ... . Написать программу, которая для заданного числа  $A$  выводит  $N$  членов последовательности Фибоначчи.

Оценка	Показатели оценки
3	Алгоритм составлен с двумя ошибками
4	Алгоритм составлен с одной ошибкой
5	Алгоритм составлен без ошибок

### Задание №11

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Напишите программу, в котором после заполнения массива случайными числами в интервале  $[0,100]$  подсчитывается отдельно среднее значение всех элементов, которые  $<50$ , и среднее значение всех элементов, которые больше или равно 50

Пример:

Массив:

4 1 51 4 61

Среднее арифметическое элементов  $[0,50)$ : 3

Среднее арифметическое элементов  $[50,100)$ : 56

Оценка	Показатели оценки
5	Алгоритм построен верно.
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков).
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №12

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Напишите программу, в котором массив заполняется случайными числами в интервале  $[-10,10]$  и отбирающий в другой массив все четные отрицательные числа.

Пример:

Массив А:

-5 6 7 -4 -6 8 -8

Массив В:

-4 -6 -8

Оценка	Показатели оценки
5	Алгоритм построен верно.
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков).
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №13

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Программа, получает два целых числа и находит их произведение, не используя операцию умножения. Учитывать, что числа могут быть и отрицательными.

*Пример:*

Введите два числа:

21 -3

$21 * (-3) = -63$

Оценка	Показатели оценки
5	Алгоритм построен верно;
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков);
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №14

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Даны целые положительные числа N и K. Используя только операции сложения и вычитания,

найти частное от деления нацело  $N$  на  $K$ , а также остаток от этого деления.

Оценка	Показатели оценки
5	Алгоритм построен верно;
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков);
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №15

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Определить, является ли заданное шестизначное число счастливым. (Счастливым называют такое шестизначное число, что сумма его первых трех цифр равна сумме его последних трех цифр.)

Оценка	Показатели оценки
5	Алгоритм построен верно;
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков);
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №16

Построить алгоритм в виде блок-схемы в онлайн-сервисе (например, [programforyou.ru/block-diagram-redactor](http://programforyou.ru/block-diagram-redactor)) для следующей задачи:

Дано натуральное число  $N$ . Если оно делится на 4, вывести на экран ответ  $N=4k$  (где  $k$  – соответствующее частное), если остаток от деления на 4 равен 1 –  $N=4k+1$ , если остаток от деления на 4 равен 2 –  $N=4k+2$ , если остаток от деления на 4 равен 3 –  $N=4k+3$ . Например:  $12=4*3$ ,  $22=4*5+2$ .

Оценка	Показатели оценки
5	Алгоритм построен верно;
4	Алгоритм построен с небольшими недочетами (например: неправильное соединение блоков);
3	Алгоритм построен с грубыми ошибками (например: неправильно изображены блоки и их действия).

### Задание №17



Определить функцию сложности алгоритма по результатам эксперимента:

N	Количество перестановок
5	62

Оценка	Показатели оценки
5	Задача решена с ошибками
4	Задача решена, есть замечания по расчетам
3	Задача решена с ошибками

### Задание №18

В среде разработки Visual Studio напишите программу, которая вычисляет сумму первых  $n$  целых положительных четных чисел. Количество суммируемых чисел вводится во время работы программы.

*Рекомендуемый вид экрана:*

```

?????????? ????? ?????????????? ?????? ??????.???????? ?????????? ???????
???? ??????:12????? ?????? 12 ?????????????? ?????? ?????? ?????? 156.

```

*(представлен один из вариантов задач)*

Оценка	Показатели оценки
3	Программа написана, но не работает из-за ошибок.
4	Программа написана, работает с незначительными ошибками.
5	Программа работает без ошибок.

### Задание №19

Написать программу в среде разработки VisualStudio, для задачи:

Напишите программу, заполняющая квадратную матрицу случайными числами в интервале  $[10,99]$ , и находит максимальный и минимальный элементы в матрице и их индексы.

Пример:

Матрица A:

12 14 67 45

32 87 45 63

69 45 14 11

40 12 35 15

Максимальный элемент  $A[2,2]=87$

Минимальный элемент  $A[3,4]=11$

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №20

Написать программу в среде разработки VisualStudio, для следующей задачи:

Даны целые положительные числа  $M$ ,  $N$ , число  $D$  и набор из  $M$  чисел. Сформировать матрицу размера  $M \times N$ , у которой первый столбец совпадает с исходным набором чисел, а элементы каждого следующего столбца равны сумме соответствующего элемента предыдущего столбца и числа  $D$  (в результате каждая строка матрицы будет содержать элементы арифметической прогрессии).

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №21

Написать программу в среде разработки VisualStudio, для следующей задачи:

Описать функцию  $\text{TimeToHMS}(T, H, M, S)$ , определяющую по времени  $T$  (в секундах) содержащееся в нем количество часов  $H$ , минут  $M$  и секунд  $S$  ( $T$  — входной,  $H$ ,  $M$  и  $S$  — выходные параметры целого типа).

Используя эту функцию, найти количество часов, минут и секунд для пяти данных отрезков времени  $T_1, T_2, \dots, T_5$ .

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №22

Написать программу в среде разработки VisualStudio, для следующей задачи:

Даны основания и высоты двух равнобедренных трапеций. Найти сумму их периметров и сумму их площадей. (Определить функцию для расчета периметра и площади равнобедренной трапеции по ее основаниям и высоте.) Площадь трапеции  $(S) = ((a + b)/2) \times h$ , где  $a$  и  $b$  — длины параллельных сторон, а  $h$  — высота трапеции. Периметр трапеции  $(P) = a + b + 2 \sqrt{h^2 + (a - b)^2 / 4}$ .

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №23

Написать программу в среде разработки VisualStudio, для следующей задачи:

Написать рекурсивную функцию для вычисления максимального элемента массива из  $n$  элементов

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №24

Написать программу в среде разработки VisualStudio, для следующей задачи:

Определить комбинированный (структурный) тип для представления информации по горным вершинам, состоящей из названия вершины и ее высоты. Ввести информацию по 5 вершинам. Вывести название самой низкой вершины из всех 5.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №25

Написать программу в среде разработки VisualStudio, для следующей задачи:

Николаю требуется проверить, возможно ли из представленных отрезков условной длины сформировать треугольник. Для этого он решил создать класс **TriangleChecker**, принимающий только положительные числа. С помощью метода **is\_triangle()** возвращаются следующие значения (в зависимости от ситуации):

- **Ура, можно построить треугольник!;**
- **С отрицательными числами ничего не выйдет!;**
- **Нужно вводить только числа!;**
- **Жаль, но из этого треугольник не сделать.**

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №26

Написать программу в среде разработки VisualStudio, для следующей задачи:

Напишите программу, выводящая фамилию и инициалы с введенных в одну строку фамилии, имени и отчество, разделенные пробелом.

Пример:

Введите фамилию, имя и отчество:

Васильев Сергей Павлович

С.П. Васильев

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №27

Создайте проект и разместите в форме с заголовком "СКОРОСТЬ" следующие компоненты, как показано на рисунке:

Тексты надписей: для Надписи1 - "РАССТОЯНИЕ", для Надписи2 - "ВРЕМЯ", для Надписи3 - "СКОРОСТЬ: " должны быть крупными и хорошо видимыми в форме. На Кнопке1 должен быть текст "ВЫЧИСЛИТЬ", на Кнопке2 – "ВЫХОД".

Текстовые поля первоначально должны быть пустыми

Для кнопки "ВЫЧИСЛИТЬ" создайте выполняющую следующие действия процедуру обработки нажатия на эту кнопку:

а) преобразование текстовых (строковых) значений свойства **Text** Текстового поля1 (для расстояния) и Текстового поля 2 (для времени) в соответствующие вещественные значения.

б) вычисление вещественного значения скорости по вещественным значениям расстояния и времени.

в) преобразование вещественного значения скорости в текстовое (строковое) значение и добавление этого значения к первоначальному тексту Надписи3 в виде текста справа.

Для кнопки "ВЫХОД" создайте процедуру обработки нажатия на эту кнопку для выхода из программы или, иначе говоря, для закрытия формы.

Отредактируйте созданную программу таким образом, чтобы она реагировала на ввод нулевого значения в Текстовое поле2 после нажатия кнопки ВЫЧИСЛИТЬ добавлением к первоначальному тексту Надписи3 текста 'ВРЕМЯ НЕ ДОЛЖНО БЫТЬ РАВНО 0!'

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

## Задание №28

Написать программу в среде разработки VisualStudio, для следующей задачи:

Напишите рекурсивную функцию `summation(n)`, которая будет принимать положительное целое число `n` и возвращать сумму чисел от 1 до `n`.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №29

Написать программу в среде разработки VisualStudio, для следующей задачи:

Напишите программу, находящая среднее арифметическое всех чисел, которые записаны в файле в столбик, и выводит результат в файл.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №30

Написать программу в среде разработки VisualStudio, для следующей задачи:

Имеется текстовый файл, в котором записана одна строка *Здравствуйте, дорогие ребята*. Приписать в конце строки восклицательный знак.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №31

Написать программу в среде разработки VisualStudio, для следующей задачи:

Определить комбинированный (структурный) тип для представления анкеты ребенка, состоящей из его имени, пола и роста. Ввести информацию по 5 детям. Вывести средний рост всех детей.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.

3	Программа написана, но не работает из-за ошибок.
---	--------------------------------------------------

### Задание №32

Написать программу в среде разработки VisualStudio, для следующей задачи:

Создайте класс **Soda** (для определения типа газированной воды), принимающий 1 аргумент при инициализации (отвечающий за добавку к выбираемому лимонаду). В этом классе реализуйте метод **show\_my\_drink()**, выводящий на печать **Газировка и {ДОБАВКА}** в случае наличия добавки, а иначе отобразится следующая фраза: **Обычная газировка.**

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №33

Написать программу в среде разработки VisualStudio, для построенного алгоритма.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №34

Написать программу в среде разработки VisualStudio, для построенного алгоритма, с использованием цикла While.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №35

Написать программу в среде разработки VisualStudio, для построенного алгоритма.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.

4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №36

Написать программу в среде разработки VisualStudio, для построенного алгоритма, с использованием оператора выбора Switch.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №37

Написать программу в среде разработки VisualStudio, для построенного алгоритма, с использованием цикла While.

Оценка	Показатели оценки
5	Программа работает верно, без ошибок.
4	Программа написана, работает с незначительными ошибками.
3	Программа написана, но не работает из-за ошибок.

### Задание №38

Оформите представленный код программы в соответствии со стандартом кодирования:

Вершины треугольника заданы массивом точек. Точки являются объектом класса CPoint. Написать функцию, вычисляющую площадь треугольника по формуле Герона:

```
#include #include class CPoint
```

```
{ private: int x, y; public:
```

```
CPoint(int _x, int _y)
```

```
(x = _x; y = _y/2;}
```



```

CPoint()

{x = 0; y = 0; } void SetX(int _x)

{x = _x;}

void SetY(int _y)

(Y = _Y; > int GetX()

{ return x; } int GetY() { return y; } };

float length(CPoint pi, CPoint p2)

{

int x1, x2, y1, y2, L; x1 = pi.GetX(); y1 = pi.GetY(); x2 = p2.GetX(); y2 = p2.GetY();

L = sqrt(pow(x2-x1, 2.0) + pow(y2-y1, 2.0)); return L;

float square(CPoint *mas)

{float L1, L2, L3, p;

L1 = length(mas[0], mas[1]);

L2 = length(mas[1], mas[2]);

L3 = length(mas[2], mas[0]); p = (L1 + L2 + L3)/2.0; if (p*(p-L1)*(p-L2)*(p-L3)>=0)

return sqrt(p*(p - L1)*(p - L2)*(p - L3)); else (printf("Треугольник построить нельзя! "); return -1;

}} int main(){

CPoint mas [3]; int x, y;

for (int i = 0; i<3; i++){printf("First point:"); scanf("%d%d", &x, &y); mas [i] .SetX(x); mas[i].SetY(y);}

printf("S = %3.2f ", square(mas)); return 0;}

```

Оценка	Показатели оценки
3	программа оформлена частично в соответствии со стандартом кодирования
4	программа оформлена в соответствии со стандартом кодирования, с небольшими недочетами
5	программа оформлена полностью в соответствии со стандартом кодирования

### Задание №39

Оформить код написанной программ в соответствии со стандартом кодирования.

Оценка	Показатели оценки
5	Код программы оформлен в соответствии со стандартом.
4	Код программы частично оформлен в соответствии со стандартом.
3	Код оформлен без соблюдения правил.

### Задание №40

Выполнить проверку и отладку кода программы.

Оценка	Показатели оценки
5	Логика программы верна, работает корректно.
4	Логика программы верна, присутствуют незначительные ошибки.
3	Программа работает не корректно.

### Задание №41

Выполните проверку и отладку следующего кода программы

Написать класс «герой». Члены класса: имя героя, его возраст. Написать класс «параметры». Члены класса: сила, защита, скорость. На базе этих двух классов написать класс «воин», включающий в себя все перечисленные члены.

Иерархия классов будет выглядеть следующим образом:

```
#include
```

```
#include
```

```
class Hero //начало определения 1-го базового класса Него
```

```
{
```

```
protected;
```

```
char name [32 ] ; //защищенная член-переменная name класса Него
```

```

int age; //защищенная член-переменная аде класса Него

public:

Него () //определение конструктора класса Него без параметров

{

strcpy(name, "NoName");

//копирование в член-переменную name текста NoName аде = 0; //присвоить члену-переменной аде
значения 0

}

Него(char* name, int age)

//определение конструктора класса Него с двумя параметрами

{

strcpy (this ->name, name) ; /*копирование члена-переменной
name (this->name,) значения параметра name V

this->age = age;

//присвоить члену-переменной age (this->age^ значение параметра age

}

void info () //определения функции-члена info

{

printf("Него:%s, %d ", name, age);

//вывод на экран названия класса и значений членов-переменных name и аде

}

}; //конец определения базового класса Него

class Parametry //начало определения базового класса Parametry

{

protected:

```

```

int sila; //защищенная член-переменная sila класса Parametry

int zashita;

//защищенная член-переменная zashita класса Parametry

int skorost;

//защищенная член-переменная skorost класса Parametr

public:

Parametry(int _sila = 0, int _zashita = 0, int _skoros = 0)

//конструктор класса Parametry со значениями поумолчаниюю

{

sila = _sila; zashita = _zashita; skorost = _skorost;

}

void info() //определение члена-функции info класса Parametry

printf("Parametry: %d, %d, %d ", sila, zashita,

skorost) ; /*вывод на экран названия класса и значения

членов-переменных sila, zashita, skorost*/

}

}; //конец определения класса Parametry

class Warrior : public Hero, Parametry

//начало определ. класса Warrior наследника классов Hero, Parametry { public:

/*определение конструктора класса Warrior инициализацией «родительских» конструкторов

классов Hero и Parametry*/

Warrior(char* _name, int _age, int _sila, int _zashita, int _skorost): Hero(_name, _age), Parametry(_sila,

_zashita, _skoros)

{ }

void info() //определение члена-функции info класса Warrior

```

```

{

printf("Warrior: ");

//вывод на экран названия класса Warrior Hero : : info () ; //вызов функции-члена info класса Hero

Parametry::info();

//вызов функции-члена info класса Parametry

}

}; //конец определения класса Warrior

int main()

Warrior w("Ivan", 30, 100, 300, 15); w.info(); return 0;

}

```

Оценка	Показатели оценки
3	проверка и отладка выполнена, программа работает с ошибками
4	проверка и отладка выполнена, программа работает с небольшими недочетами
5	проверка и отладка выполнена, программа работает верно