

Министерство науки и высшего образования Российской Федерации  
**Муромский институт (филиал)**  
федерального государственного бюджетного образовательного учреждения высшего образования  
**«Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»**  
(МИ ВлГУ)

Кафедра *ПИИ*

«УТВЕРЖДАЮ»  
Заместитель директора по УР  
\_\_\_\_\_ Д.Е. Андрианов  
\_\_\_\_\_ 16.06.2020

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

*Основы алгоритмизации и программирования*

**Направление подготовки**

*09.03.04 Программная инженерия*

**Профиль подготовки**

Семестр	Трудоем- кость, час./зач. ед.	Лек- ции, час.	Практи- ческие занятия, час.	Лабора- торные работы, час.	Консуль- тация, час.	Конт- роль, час.	Всего (контак- тная работа), час.	СРС, час.	Форма промежу- точного контроля (экз., зач., зач. с оц.)
<b>1</b>	<b>216 / 6</b>	<b>32</b>		<b>48</b>	<b>5,2</b>	<b>0,35</b>	<b>85,55</b>	<b>94,8</b>	<b>Экз.(35,65)</b>
<b>Итого</b>	<b>216 / 6</b>	<b>32</b>		<b>48</b>	<b>5,2</b>	<b>0,35</b>	<b>85,55</b>	<b>94,8</b>	<b>35,65</b>

Муром, 2020 г.

## 1. Цель освоения дисциплины

Целью дисциплины является изучение и освоение базовых понятий и приемов программирования, применяемых на всех основных этапах разработки программ; изучение методов программирования для овладения знаниями в области технологии программирования; подготовка к осознанному использованию как языков программирования, так и методов программирования.

Задачами дисциплины являются формирование целостного представления о предмете, освоение теоретических знаний и практических навыков, позволяющих ориентироваться в области разработки алгоритмов решения задач и написания программных кодов на языке высокого уровня; освоение теоретических основ структурного программирования; изучение конкретных языков программирования; использования языков программирования для решения практических задач.

## 2. Место дисциплины в структуре ОПОП ВО

Дисциплина базируется на школьном курсе информатики и математики, и является базовой для всех специальных дисциплин направления подготовки.

## 3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов;	ОПК-6.1 Использует современные языки, утилиты и среды программирования	<b>Знать</b> основные алгоритмические конструкции процедурного программирования (ОПК-6.1)  <b>Уметь</b> составлять алгоритмы решения задач (ОПК-6.1)  <b>Владеть</b> навыками разработки алгоритмов решения практических задач обработки данных (ОПК-6.1)	Тест, задания на лабораторную работу

#### 4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 6 зачетных единиц, 216 часов.

##### 4.1. Форма обучения: очная

Уровень базового образования: среднее общее.

Срок обучения 4г.

##### 4.1.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Основы алгоритмизации	1	2		4					62	отчет по лабораторным работам
2	Основы программирования	1	22		20					18	тестирование, отчет по лабораторным работам
3	Решение практических задач	1	8		24					14,8	отчет по лабораторным работам
Всего за семестр		216	32		48			5,2	0,35	94,8	Экз.(35,65)
Итого		216	32		48			5,2	0,35	94,8	35,65

##### 4.1.2. Содержание дисциплины

###### 4.1.2.1. Перечень лекций

###### Семестр 1

*Раздел 1. Основы алгоритмизации*

###### Лекция 1.

Понятие алгоритма (2 часа).

*Раздел 2. Основы программирования*

###### Лекция 2.

Введение в язык программирования Си (2 часа).

###### Лекция 3.

Условный оператор и логическое выражение на языке Си (2 часа).

###### Лекция 4.

Операторы цикла в языке Си (2 часа).

###### Лекция 5.

Одномерные массивы (2 часа).

###### Лекция 6.

Основные алгоритмы обработки массивов (2 часа).

###### Лекция 7.

Функции (2 часа).

###### Лекция 8.

Рекурсия (2 часа).

###### Лекция 9.

Строки в языке Си (2 часа).

**Лекция 10.**

Двумерные массивы на языке Си (2 часа).

**Лекция 11.**

Основные алгоритмы обработки двумерных массивов (2 часа).

**Лекция 12.**

Структуры в языке Си (2 часа).

**Раздел 3. Решение практических задач****Лекция 13.**

Обработка ошибок в Си (2 часа).

**Лекция 14.**

Преобразование данных в языке Си. Оператор "?". Оператор "." (2 часа).

**Лекция 15.**

Работа с файлами на языке Си (2 часа).

**Лекция 16.**

Многомодульные программы (2 часа).

**4.1.2.2. Перечень практических занятий**

Не планируется.

**4.1.2.3. Перечень лабораторных работ****Семестр 1****Раздел 1. Основы алгоритмизации****Лабораторная 1.**

Составление алгоритмов решения задачи (4 часа).

**Раздел 2. Основы программирования****Лабораторная 2.**

Линейные программы на языке Си (4 часа).

**Лабораторная 3.**

Условный оператор в языке Си (4 часа).

**Лабораторная 4.**

Операторы цикла в языке Си (4 часа).

**Лабораторная 5.**

Одномерные статические массивы (4 часа).

**Лабораторная 6.**

Написание функций на языке Си (4 часа).

**Раздел 3. Решение практических задач****Лабораторная 7.**

Обработка строк в языке Си (4 часа).

**Лабораторная 8.**

Обработка двумерных массивов на языке Си (4 часа).

**Лабораторная 9.**

Случайные числа и метод Монте-Карло (4 часа).

**Лабораторная 10.**

Работа со структурами на языке Си (4 часа).

**Лабораторная 11.**

Чтение и запись файлов (4 часа).

**Лабораторная 12.**

Многомодульные программы (4 часа).

**4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы**

Перечень тем, вынесенных на самостоятельное изучение:

1. Способы записи алгоритмов.
2. Языки программирования. Эволюция языков программирования.

3. Классификация языков программирования.
4. Интегрированная среда программирования.
5. Методы программирования: структурный, модульный, объектно-ориентированный.
6. Достоинства и недостатки методов программирования.
7. Общие принципы разработки программного обеспечения.
8. Жизненный цикл программного обеспечения.
9. Типы приложений.
10. Консольные приложения.
11. Файлы.
12. Файловые переменные.
13. Стандартные (текстовые и бестиповые) и типизированные файлы.
14. Технологические цепочки обработки файлов различных типов.
15. Подпрограммы для работы с файлами различных типов.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

#### **4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР**

Не планируется.

#### **4.1.2.6. Примерный перечень тем курсовых работ (проектов)**

Не планируется.

## 4.2 Форма обучения: заочная

Уровень базового образования: среднее общее.

Срок обучения 5л.

Семестр	Трудоем- кость, час./ зач. ед.	Лек- ции, час.	Практи- ческие занятия, час.	Лабора- торные работы, час.	Консультация, час.	Конт- роль, час.	Всего (контакт- ная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
1	103 / 2,86	10		12	5	0,5	27,5	71,75	Зач.(3,75)
2	113 / 3,14	10		20	5	0,6	35,6	68,75	Экз.(8,65)
Итого	216 / 6	20		32	10	1,1	63,1	140,5	12,4

### 4.2.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Этапы решения задач на ЭВМ. Критерии качества программы. Жизненный цикл программы	1	4							23	тестирование
2	Способы записи алгоритма	1	6							29	тестирование
3	Процесс программирования	1			12					19,75	тестирование, отчет по лабораторным работам
Всего за семестр		103	10		12	+		5	0,5	71,75	Зач.(3,75)
4	Процесс программирования	2	10		20					68,75	Устный опрос
Всего за семестр		113	10		20	+		5	0,6	68,75	Экз.(8,65)
Итого		216	20		32			10	1,1	140,5	12,4

### 4.2.2. Содержание дисциплины

#### 4.2.2.1. Перечень лекций

##### Семестр 1

Раздел 1. Этапы решения задач на ЭВМ. Критерии качества программы. Жизненный цикл программы

##### Лекция 1.

Основные этапы решения задач на ЭВМ, критерии качества программы (2 часа).

## **Лекция 2.**

Жизненный цикл программы. Постановка задачи и спецификация программы (2 часа).

### *Раздел 2. Способы записи алгоритма*

## **Лекция 3.**

Анализ постановки задачи и ее предметной области. Формальное решение задачи.

Алгоритмизация (2 часа).

## **Лекция 4.**

Способы записи алгоритма. Визуальные алгоритмы. Представление основных управляющих структур программирования (2 часа).

## **Лекция 5.**

Теорема структуры и структурное программирование (2 часа).

## **Семестр 2**

### *Раздел 4. Процесс программирования*

## **Лекция 6.**

Процесс программирования: Программа на языке высокого уровня. Основные понятия языков программирования. Язык программирования C++ (2 часа).

## **Лекция 7.**

Процесс программирования: Способы конструирования и верификации программ.

Среда программирования Visual Studio. Событийная модель ОС Windows (2 часа).

## **Лекция 8.**

Процесс программирования: Концепция объектно-ориентированного программирования (2 часа).

## **Лекция 9.**

Процесс программирования: Анализ программ. Утверждения о программах.

Корректность программ. Создание программы в Visual Studio (2 часа).

## **Лекция 10.**

Процесс программирования: Принципы визуального программирования. Свойства и события. Основные понятия алгоритмического языка. Комментарии в программе. Директивы компилятора (2 часа).

### **4.2.2.2. Перечень практических занятий**

Не планируется.

### **4.2.2.3. Перечень лабораторных работ**

## **Семестр 1**

### *Раздел 1. Процесс программирования*

## **Лабораторная 1.**

Программирование линейных алгоритмов (4 часа).

## **Лабораторная 2.**

Программирование разветвляющихся алгоритмов (4 часа).

## **Лабораторная 3.**

Массивы (4 часа).

## **Семестр 2**

### *Раздел 2. Процесс программирования*

## **Лабораторная 4.**

Функции (4 часа).

## **Лабораторная 5.**

Создание GUI на Visual C# (4 часа).

## **Лабораторная 6.**

Диалоговые окна (4 часа).

## **Лабораторная 7.**

Работа с компонентом Chart (4 часа).

## **Лабораторная 8.**

Изучение вероятностных алгоритмов (4 часа).

#### **4.2.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы**

Перечень тем, вынесенных на самостоятельное изучение:

1. Критерии качества программы. Жизненный цикл программы.
2. Постановка задачи и спецификация программы.
3. Составные части программного проекта.
4. Языки структурного программирования.
5. Корректность программ.
6. Утверждения о программах.
7. Способы описания подпрограмм. Передача параметров.
8. Варианты представления массивов. Утверждения о массивах.
9. Индуктивные функции на последовательностях (файлах, массивах).
10. Программирование динамических структур в ЯВУ.
11. Организация диалогов.
12. Способы конструирования и верификации программ.
13. Надежность программного средства. Источники ошибок в программных средствах.

Обеспечение надежности.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

#### **4.2.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР**

1. Примерные задания на контрольную работу:
2. Напишите программу на С, которая находит максимальный и минимальный элемент в массиве.
3. Напишите программу на С для разделения нечетных и четных целых чисел в отдельные массивы.
4. Напишите программу на С для сортировки элементов массива в порядке возрастания.
5. Напишите программу на С для вычитания двух матриц.
6. Напишите программу на С, чтобы найти транспонирование заданной матрицы.
7. Напишите программу на С для вычисления определителя матрицы 3x3.
8. Напишите программу на С, чтобы найти самый большой подмассив с равным количеством нулей и единиц.
9. Напишите программу на С для проверки четности или нечетности заданного числа с помощью функции.
10. Напишите программу на С, чтобы проверить, является ли число простым или нет, используя функцию.
11. Напишите программу на С, чтобы найти сумму ряда  $1!/1+2!/2+3!/3+4!/4+5!/5$  с помощью функции.
12. Напишите программу на С, чтобы проверить, является ли число простым или нет, используя функцию.
13. Напишите программу на С для проверки армстронговских и совершенных чисел с помощью функции.
14. Напишите на С программу, которая считывает с клавиатуры 10 чисел и находит их сумму и среднее значение.
15. Напишите программу на С, которая отображает куб числа для заданного целого числа.
16. Напишите на С программу, которая считывает с клавиатуры 10 чисел и находит их сумму и среднее значение.
17. Напишите программу на С для отображения таблицы умножения заданного целого числа.

#### **4.2.2.6. Примерный перечень тем курсовых работ (проектов)**

Не планируется.



## **5. Образовательные технологии**

В процессе изучения дисциплины применяется контактная технология преподавания (за исключением самостоятельно изучаемых студентами вопросов). При проведении лабораторных работ применяется имитационный или симуляционный подход, когда преподавателем разбирается на конкретном примере проблемная ситуация, и демонстрируются при помощи мультимедийной техники шаги решения. Затем студенты самостоятельно решают аналогичные задания.

Во время выполнения лабораторных работ каждому студенту выдается конкретное задание, тем самым формируется способность обучающихся к самостоятельной работе при решении определенных задач, связанных с изучаемой темой.

## **6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.**

Фонды оценочных материалов (средств) приведены в приложении.

## **7. Учебно-методическое и информационное обеспечение дисциплины.**

### **7.1. Основная учебно-методическая литература по дисциплине**

1. Керниган, Б. В. Язык программирования C : учебник / Б. В. Керниган, Д. М. Ричи. — 3-е изд. — Москва : ИНТУИТ, Ай Пи Ар Медиа, 2021. — 313 с. — URL: <https://www.iprbookshop.ru/102075.html>

2. Двойнишников, С. В. Основы программирования (язык C) : учебное пособие / С. В. Двойнишников, К. Ф. Лысаков. — Новосибирск : НГУ, 2022. — 138 с. — URL: <https://www.iprbookshop.ru/128151.html>

### **7.2. Дополнительная учебно-методическая литература по дисциплине**

1. Шульга, Т. Э. Основы программирования на языке C : учебное пособие / Т. Э. Шульга. — Саратов : Саратовский государственный технический университет имени Ю.А. Гагарина, ЭБС АСВ, 2012. — 83 с. — ISBN 978-5-7433-2662-4. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/76494.html> (дата обращения: 18.10.2021). — Режим доступа: для авторизир. пользователей. - DOI: <https://doi.org/10.23682/76494> - <https://www.iprbookshop.ru/76494.html>

### **7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем**

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института ([www.mivlgu.ru/iop](http://www.mivlgu.ru/iop)), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

Информационно-образовательный портал МИ ВлГУ - <https://www.mivlgu.ru/iop/>

Электронная библиотечная система iprBooks.ru - <http://www.iprbooks.ru>

Электронная библиотека «ЭВРИКА» - <https://evrika.mivlgu.ru/>

Электронная библиотека ВлГУ - <https://dspace.www1.vlsu.ru/>

Программное обеспечение:

LibreOffice (Mozilla Public License v2.0)

Notepad++ (GNU GPL 3)  
Microsoft Visual Studio

#### **7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

iprbookshop.ru  
mivlgu.ru  
iprbooks.ru  
mivlgu.ru/iop

### **8. Материально-техническое обеспечение дисциплины**

Лаборатория программного обеспечения и сопровождения компьютерных систем  
Сервер «Ай Тек» на базе 2 процессоров Intel Xeon; 12 шт. компьютеров Intel Core i5-10400 2,9 GHz/ 8Gb DDR-4/ SSD-480 Gb/ Hiper 21,5'; интерактивная доска SMART Board 480 со встроенным проектором V25; маршрутизатор Gigabit Switch TEG-S16S. Маркерная доска. Доступ к сети Интернет.

### **9. Методические указания по освоению дисциплины**

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями.

До выполнения лабораторных работ обучающийся изучает соответствующий раздел теории. Перед занятием студент знакомится с описанием заданий для выполнения работы, внимательно изучает содержание и порядок проведения лабораторной работы. Лабораторная работа проводится в компьютерном классе. Обучающиеся выполняют индивидуальную задачу компьютерного моделирования в соответствии с заданием на лабораторную работу. Полученные результаты исследований сводятся в отчет и защищаются по традиционной методике в классе на следующем лабораторном занятии. Необходимый теоретический материал, индивидуальное задание, шаги выполнения лабораторной работы и требование к отчету приведены в методических указаниях, размещенных на информационно-образовательном портале института.

Самостоятельная работа оказывает важное влияние на формирование личности будущего специалиста, она планируется обучающимся самостоятельно. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Форма заключительного контроля при промежуточной аттестации – экзамен. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению  
*09.03.04 Программная инженерия*  
Рабочую программу составил *к.т.н., доцент кафедры ПИН Привезенцев*  
*Д.Г.*\_\_\_\_\_

Программа рассмотрена и одобрена на заседании кафедры *ПИН*  
протокол № 8 от 15.05.2020 года.  
Заведующий кафедрой *ПИН* \_\_\_\_\_ *Жизняков А.Л.*  
(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической  
комиссии факультета  
протокол № 10 от 10.06.2020 года.  
Председатель комиссии ФИТР \_\_\_\_\_ *Рыжкова М.Н.*  
(Подпись) (Ф.И.О.)

**Фонд оценочных материалов (средств) по дисциплине**  
**Основы алгоритмизации и программирования**

## **1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине**

Задания для выполнения лабораторных работ приведены в методических указаниях.

Примеры задач:

Составить блок-схему алгоритма решения задачи и разработать программу на языке высокого уровня реализующую алгоритм решения:

- Определить количество целых чисел, входящих в массив только один раз
- Подсчитать количество вхождений каждого числа в массив
- Дана строка символов. Подсчитать количество слов и количество символов во всех словах, состоящих только из цифр
- Дана строка символов, состоящая из символов "0"-"9", "A"-"F". Рассматривая данную последовательность как запись числа в шестнадцатеричной системе счисления, составить алгоритм и программу перевода его числовую форму
- В матрице поменять местами строку с max элементом и строку с min элементом
- В матрице найти сумму элементов строки, в которой расположен элемент с наименьшим значением

### **Общее распределение баллов текущего контроля по видам учебных работ для студентов**

Рейтинг-контроль 1	отчеты по 3 лабораторным работам	до 15
Рейтинг-контроль 2	отчеты по 4 лабораторным работам	до 15
Рейтинг-контроль 3	отчеты по 4 лабораторным работам	до 15
Посещение занятий студентом		до 5
Дополнительные баллы (бонусы)	Активность на занятиях	до 5
Выполнение семестрового плана самостоятельной работы		до 5

## **2. Промежуточная аттестация по дисциплине**

**Перечень вопросов к экзамену / зачету / зачету с оценкой.**

**Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)**

Оценочные средства для промежуточной аттестации в виде итогового теста приведены на информационно-образовательном портале по ссылке

<https://www.mivlgu.ru/iop/course/view.php?id=2082>

### **Методические материалы, характеризующие процедуры оценивания**

Из комплекта тестовых заданий программным комплексом информационно-образовательного портала МИ ВлГУ формируются в автоматическом режиме индивидуальный вариант тестирования студента. Программный комплекс формирует индивидуальные задания для каждого зарегистрированного в системе студента и контролирует время прохождения тестирования. Результатом тестирования является процент правильных ответов. Итоговая

оценка по предмету складывается из результатов тестирования и индивидуального семестрового рейтинга студента.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	Уровень сформированности компетенций
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	<b>Высокий уровень</b>
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	<b>Продвинутый уровень</b>
50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	<b>Пороговый уровень</b>
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	<b>Компетенции не сформированы</b>

### 3. Задания в тестовой форме по дисциплине

Примеры заданий:

Параметры, указываемые при вызове функции (метода), называются ...

- фактическими параметрами
- формальными параметрами

- задаваемыми параметрами
- записываемыми параметрами

Начальное значение локальных переменных подпрограммы в языке C/C++ ...

- неопределенно
- равно нулю
- равно случайному значению
- пустое

В языке C/C++ символ языка for определяет оператор цикла с ... (параметром)

Найти значение c для заданных a=2,b=4,c=3,n=2:

```
for (int i:= 0; i< n; i++) c = c + a * b;
```

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/question/edit.php?courseid=2082&deleteall=1&category=54222%2C61242&qbshowtext=1&recurse=0&recurse=1&showhidden=0>

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.