

«

»

“ ”

“ ”

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ  
Программирование и алгоритмизация**

: 13.03.02

, :

: 2, : 3

		<b>3</b>
<b>1</b>	( )	4
<b>2</b>		144
<b>3</b>	, .	65
<b>4</b>	, .	18
<b>5</b>	, .	0
<b>6</b>	, .	36
<b>7</b>	, .	0
<b>8</b>	, .	2
<b>9</b>	, .	9
<b>10</b>	, .	79
<b>11</b>	( , , )	.
<b>12</b>		

( ): 13.03.02

955 03.09.2015 ., : 25.09.2015 .

: 1, ,

( ): 13.03.02

,  
,

6 20.06.2017  
4 20.06.2017

, 6 21.06.2017

:

, . . . . . . . .

:

, . . . . . . . .  
, . . . . . . . .

:

. . .

# 1.

1.1

<b>Компетенция ФГОС: ОПК.1</b> способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий; <i>в части следующих результатов обучения:</i>	
4.	,
5.	
<b>Компетенция ФГОС: ПК.1</b> способность участвовать в планировании, подготовке и выполнении типовых экспериментальных исследований по заданной методике; <i>в части следующих результатов обучения:</i>	
2.	

# 2.

2.1

	(	
--	---	--

<b>.1. 4</b>	,
1.34. знать основные команды и операторы языка высокого уровня, основные концепции и понятия программирования	; ;
<b>.1. 5</b>	
2.у5. уметь использовать специализированные программные средства при решении профессиональных задач	; ;
<b>.1. 2</b>	
3.уб. уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	; ;

# 3.

3.1

	,	.		
<b>: 3</b>				
:				

<p>1. 1.</p> <p>( : ), ,</p> <p>( - )</p> <p>( ).</p> <p>-</p>	0	4	1	-
<p>2. 2.</p> <p>( ).</p>	0	4	1, 3	switch. if
<p>3. 3.</p> <p>-</p>	0	2	1, 3	do. for while
:				

<p>4. 4.</p> <p>PROLOG.</p> <p>DOS IDE</p> <p>Visual Studio.</p>	0	4	1	
<p>5. 5.</p> <p>Visual-Basic.</p>	0	4	1, 2	Visual-Basic.

: 3				
:				

1. 1.	Visual Basic.	0	16	2, 3	Visual Basic.
:					
2. 2.		0	12	1, 2, 3	:
3. 3.	MathCad.	0	8	2, 3	MathCad. MathCad.

3.3

:					
<b>: 3</b>					
:					
1. 1.	Acces.	0	29	2, 3	Acces.
:					
2.	Visual-Basi	0	18	2, 3	Visual-Basic.

4.

:					
<b>: 3</b>					
1			1, 2, 3	10	0

<p>: [ ]:  , [2013]. - : <a href="http://elibrary.nstu.ru/source?bib_id=vtls000179564">http://elibrary.nstu.ru/source?bib_id=vtls000179564</a>.  [2016]. - : <a href="http://elibrary.nstu.ru/source?bib_id=vtls000233408">http://elibrary.nstu.ru/source?bib_id=vtls000233408</a>. -</p>				
2		1	8	3
<p>: [ ]:  , [2013]. - : <a href="http://elibrary.nstu.ru/source?bib_id=vtls000179564">http://elibrary.nstu.ru/source?bib_id=vtls000179564</a>.  [2016]. - : <a href="http://elibrary.nstu.ru/source?bib_id=vtls000233408">http://elibrary.nstu.ru/source?bib_id=vtls000233408</a>. -</p>				
3		1, 2, 3	14	6
<p>: [ ]:  , [2013]. - : <a href="http://elibrary.nstu.ru/source?bib_id=vtls000179564">http://elibrary.nstu.ru/source?bib_id=vtls000179564</a>.  [2016]. - : <a href="http://elibrary.nstu.ru/source?bib_id=vtls000233408">http://elibrary.nstu.ru/source?bib_id=vtls000233408</a>. -</p>				
4		2, 3	47	0
<p>3.3 :  [ ]:  : <a href="http://elibrary.nstu.ru/source?bib_id=vtls000179564">http://elibrary.nstu.ru/source?bib_id=vtls000179564</a>. -  [ ]:  , [2016]. - :  <a href="http://elibrary.nstu.ru/source?bib_id=vtls000233408">http://elibrary.nstu.ru/source?bib_id=vtls000233408</a>. -</p>				

## 5.

(. 5.1).

5.1

	-
	: <a href="http://ciu.nstu.ru/kaf/persons/20205">http://ciu.nstu.ru/kaf/persons/20205</a> ; : <a href="http://vk.com/id92050604">http://vk.com/id92050604</a> ;
	: <a href="http://vk.com/id92050604">http://vk.com/id92050604</a>
	: <a href="http://dispace.edu.nstu.ru/didesk/course/show/1606/1">http://dispace.edu.nstu.ru/didesk/course/show/1606/1</a>

5.2

1	
<p><b>Краткое описание применения:</b> Задача по рассматриваемой теме выносится на рассмотрение студентам для поиска коллективного решения. Ответ обсуждается и обосновывается. Стимулирующим ресурсом к участию являются баллы.</p>	



1. ЭБС НГТУ : <http://elibrary.nstu.ru/>
2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>
3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>
4. ЭБС "Znaniium.com" : <http://znaniium.com/>
5. :

## 8.

### 8.1

1. Мятеж А. В. Программирование и алгоритмизация [Электронный ресурс] : электронный учебно-методический комплекс / А. В. Мятеж ; Новосиб. гос. техн. ун-т. - Новосибирск, [2016]. - Режим доступа: [http://elibrary.nstu.ru/source?bib\\_id=vtls000233408](http://elibrary.nstu.ru/source?bib_id=vtls000233408). - Загл. с экрана.

### 8.2

1 Visual Studio

## 9.

1	11	

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Новосибирский государственный технический университет»

Кафедра электротехнических комплексов

“УТВЕРЖДАЮ”  
ДЕКАН ФМА  
к.т.н., доцент М.Е. Вильбергер  
“ \_\_\_ ” \_\_\_\_\_ г.

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

### УЧЕБНОЙ ДИСЦИПЛИНЫ

#### Программирование и алгоритмизация

Образовательная программа: 13.03.02 Электроэнергетика и электротехника, профиль:  
Электротехника, электромеханика и электротехнологии

## 1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине Программирование и алгоритмизация приведена в Таблице.

Таблица

Формируемые компетенции	Показатели сформированности компетенций (знания, умения, навыки)	Темы	Этапы оценки компетенций	
			Мероприятия текущего контроля (курсовой проект, РГЗ(Р) и др.)	Промежуточная аттестация (экзамен, зачет)
ОПК.1 способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий	34. знать основные команды и операторы языка высокого уровня, основные концепции и понятия программирования	1. Алгоритм и его свойства. Способы записи алгоритмов. Понятие алгоритма. Свойства алгоритма: детерминированность (определенность), дискретность, конечность, результативность, массовость. Словесно-формульное (вербальное) представление алгоритма. Блок-схемы алгоритмов. Универсальный алгоритмический язык (псевдокод). Данные алгоритмов - константы и переменные. Идентификаторы. Сложные типы данных - массивы и структуры. 2. Двумерные массивы. Функции, процедуры. Работа с внешними файлами. Использование нескольких программных модулей. Формы. Элементы управления. Меню. 2. Линейная алгоритмическая структура. Последовательное выполнение действий. Изображение на блок-схеме и запись на псевдокоде. Операции присваивания, ввода и вывода данных, обращения к вспомогательному алгоритму. Разветвляющаяся алгоритмическая структура. Бинарное ветвление (альтернатива). Обход. Изображение на блок-схеме и запись на псевдокоде. Условия ветвления, формирование сложных условий. Множественный выбор. Реализация множественного выбора как последовательности бинарных. 3. Циклические алгоритмические структуры. Цикл по переменной. Цикл с постусловием и цикл с предусловием. Изображение	Контрольные работы, разделы 2-5	Зачет, вопросы 1-20

		на блок-схеме.		
ОПК.1	у5. уметь использовать специализированные программные средства при решении профессиональных задач	2. Двумерные массивы. Функции, процедуры. Работа с внешними файлами. Использование нескольких программных модулей. Формы. Элементы управления. Меню. 3. Введение в MathCad. 5. Модели данных в информационных системах. Реляционная модель базы данных. Свойства реляционных таблиц. Основные и вспомогательные таблицы (справочники). Связи таблиц: один к одному, один ко многим, многие ко многим. Первая, вторая и третья нормальные формы. Основные операции с данными. Создание таблиц в режиме конструктора. Задание свойств полей. Ввод данных в режиме таблицы и с использованием форм ввода. Конструирование запросов. Запись условий отбора.		Зачет, вопросы 1-20
ПК.1/НИ способность участвовать в планировании, подготовке и выполнении типовых экспериментальных исследований по заданной методике	у2. уметь использовать ЭВМ при имитационном моделировании заданного исследуемого процесса	Разработка программы в среде программирования Visual-Basic 1. Основы Acces. Формы. Элементы управления. Меню. 1. Основы работы в интегрированной среде программирования Visual Basic. Работа с отладчиком. Программирование простейших задач с использованием линейных структур. Программирование простейших задач с использованием структур ветвления. Программирование задач с использованием циклов, связанных с обработкой одномерных массивов и строковой информации. 2. Двумерные массивы. Функции, процедуры. Работа с внешними файлами. Использование нескольких программных модулей. Формы. Элементы управления. Меню. 3. Введение в MathCad.	Контрольные работы, разделы 1-5	Зачет, вопросы 21-38

## 2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по дисциплине проводится в 3 семестре - в форме дифференцированного зачета, который направлен на оценку сформированности компетенций ОПК.1, ПК.1/НИ.

Зачет проводится в письменной форме по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-20, второй вопрос из диапазона вопросов 21-38 (список вопросов приведен ниже). В ходе зачета преподаватель вправе задавать студенту дополнительные вопросы из общего перечня.

Кроме того, сформированность компетенции проверяется при проведении мероприятий текущего контроля, указанных в таблице раздела 1.

В 3 семестре обязательным этапом текущей аттестации является контрольная работа. Требования к выполнению контрольной работы, состав и правила оценки сформулированы в паспорте контрольной работы.

Общие правила выставления оценки по дисциплине определяются балльно-рейтинговой системой, приведенной в рабочей программе учебной дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенции ОПК.1, ПК.1/НИ за которые отвечает дисциплина, на разных уровнях.

### **Общая характеристика уровней освоения компетенций.**

**Ниже порогового.** Уровень выполнения работ не отвечает большинству основных требований, теоретическое содержание курса освоено частично, пробелы могут носить существенный характер, необходимые практические навыки работы с освоенным материалом сформированы не достаточно, большинство предусмотренных программой обучения учебных заданий не выполнены или выполнены с существенными ошибками.

**Пороговый.** Уровень выполнения работ отвечает большинству основных требований, теоретическое 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 виды заданий выполнены с ошибками.

**Базовый.** Уровень выполнения работ отвечает всем основным требованиям, теоретическое 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 из выполненных заданий, возможно, содержат ошибки.

**Продвинутый.** Уровень выполнения работ отвечает всем требованиям, теоретическое 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.

## Паспорт зачета

по дисциплине «Программирование и алгоритмизация», 3 семестр

### 1. Методика оценки

Зачет проводится в письменной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-20, второй вопрос из диапазона вопросов 21-38 (список вопросов приведен ниже). В ходе зачета преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

### Форма билета для зачета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
Факультет ФМА

#### Билет №1

к зачету по дисциплине «Программирование и алгоритмизация»

---

1. Что такое «вложенные циклы»?
2. Этапы решения задач с помощью ЭВМ.

Утверждаю: зав. кафедрой ЭТК \_\_\_\_\_ д.т.н., профессор, Щуров Н.И.  
(подпись) (дата)

### 2. Критерии оценки

- Ответ на билет к зачету считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, оценка менее 10 баллов.
- Ответ на билет к зачету засчитывается на **пороговом** уровне, если студент дает определение основных понятий, оценка составляет 11 - 14 балла.
- Ответ на билет к зачету засчитывается на **базовом** уровне, если студент формулирует основные понятия, законы, дает характеристику процессов, явлений, может представить качественные характеристики процессов, оценка составляет 15 – 17 баллов.
- Ответ на билет к зачету засчитывается на **продвинутом** уровне, если студент способен системно представлять решение задачи, давать количественные характеристики определенных процессов, приводить конкретные примеры из практики, оценка

составляет 18 - 20 баллов.

### 3. Шкала оценки

Итоговая оценка в баллах по дисциплине составляется из суммы баллов, полученных за участие на лекционных занятиях (максимум 30 баллов), выполнение и защиту лабораторных работ (максимум 30 баллов), выполнение контрольной работы (максимум 20 баллов) и сдачу зачета (максимум 20 баллов) в письменной форме.

Диапазон баллов рейтинга	Оценка ECTS	Традиционная (4-уровневая) шкала оценки	
98-100	A+	отлично	зачтено
93-97	A		
90-92	A-		
87-89	B+		
83-86	B	хорошо	
80-82	B-		
77-79	C+		
73-76	C		
70-72	C-	удовл.	
67-69	D+		
63-66	D		
60-62	D-		
50-59	E	неуд.	незачтено
25-49	FX		
0-24	F		

В общей оценке по дисциплине баллы на зачете учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

### 4. Вопросы к зачету по дисциплине «Программирование и алгоритмизация»

1. Понятие «алгоритм».
2. Что такое «исполнитель алгоритма»?
3. Свойства алгоритма.
4. Форма записи алгоритмов.
5. Графический способ записи алгоритмов.
6. Что такое «псевдокод»? (алгоритмический язык)
7. Базовые алгоритмические структуры.
8. Пример записи алгоритма на алгоритмическом языке с использованием трех базовых структур: следование, ветвление, цикл.
9. Какие циклы называют итерационными?
10. Что такое «вложенные циклы»?
11. Языки программирования.
12. Что такое «уровень языка программирования»?
13. Машинные языки, их достоинства и недостатки.
14. Язык ассемблера.
15. В чем преимущества алгоритмических языков перед машинными?
16. Какие компоненты образуют алгоритмический язык?
17. Какие понятия используют алгоритмические языки?
18. Что такое «стандартная функция»?

19. Форма записи арифметических выражений.
20. Форма записи логических выражений.
21. Этапы решения инженерной задачи с помощью ПЭВМ, их краткая характеристика. Детальное изучение содержания этапов постановки задачи, математического формулирования и математического моделирования.
22. Примеры математических моделей физических процессов и из предметной области. Методы решения. Погрешности. Способы устранения погрешностей.
23. Понятие алгоритма. Структуры алгоритмов. Типы вычислительных алгоритмов. Свойства алгоритмов, способы записи.
24. Кодирование алгоритма, программа. Структурный подход к разработке алгоритмов. Типы и характеристика основных структур, структурированное программирование. Языки программирования.
25. Общие свойства рабочих языков программирования. Основные составляющие любого ЯВУ: алфавит, синтаксис и семантика, типы данных. Языки как средства описания алгоритмов.
26. Технологии программирования. Пошаговый метод разработки алгоритма. Пример пошаговой разработки с выделением структур, отладка и тестирование алгоритма. Методы отыскания ошибок и безошибочного программирования. Средства программирования.
27. Понятие о сетях ЭВМ. Назначение и структурные схемы сетей. Internet. Особенности организации ЛВС. Топология ЛВС. Протоколы передачи информации, информационных технологий на сетях, основы телекоммуникаций и распределенной обработки информации.
28. Основы защиты информации и сведений, составляющих государственную тайну; способы хищения информации, методы защиты информации.
29. Электронная подпись.
30. Этапы решения задач с помощью ЭВМ.
31. Стили программирования.
32. Структурирование программного продукта.
33. Функциональное программирование.
34. Объектно-ориентированное программирование.
35. Базы данных.
36. Сортировка.
37. Библиотека программ.
38. Рекурсия. Рекурсивные алгоритмы.

## Паспорт контрольной работы

по дисциплине «Программирование и алгоритмизация», 3 семестр

### 1. Методика оценки

В рамках контрольной работы студентам предлагается разработать и написать на языке высокого уровня программу, которую студент выбирает самостоятельно по согласованию с преподавателем.

Обязательные структурные части контрольной работы:

1. Задание на выполнение работы;
2. Выбор способа решения и создание математической модели к поставленному заданию (максимально 4 балла);
3. Написание алгоритма программы под математическую модель (максимально 4 балла);
4. Разработка экранных форм для выполнения проекта (максимально 4 балла);
5. Написание и отладка программного кода на языке высокого уровня (максимально 4 балла);
6. Приведение контрольного примера, максимально иллюстрирующего адекватность работы программы и словесное описание ее кода (максимально 4 балла).

Оцениваемые позиции:

- полнота раскрытия вопроса;
- точность ответа;
- последовательность;
- наличие примеров.

### 2. Критерии оценки

- Работа считается **не выполненной**, если выполнены не все части контрольной работы, отсутствует анализ объекта, диагностические признаки не обоснованы, аппаратные средства не выбраны или не соответствуют современным требованиям, оценка менее 10 баллов.
- Работа считается выполненной **на пороговом** уровне, если части контрольной работы выполнены формально: анализ объекта выполнен без декомпозиции, диагностические признаки недостаточно обоснованы, аппаратные средства не соответствуют современным требованиям, оценка составляет 10-13 баллов.
- Работа считается выполненной **на базовом** уровне, если анализ объекта выполнен в полном объеме, признаки и параметры диагностирования обоснованы, алгоритмы разработаны, но не оптимизированы, аппаратные средства выбраны без достаточного обоснования, оценка составляет 14-17 баллов.
- Работа считается выполненной **на продвинутом** уровне, если анализ объекта выполнен в полном объеме, признаки и параметры диагностирования обоснованы, алгоритмы разработаны и оптимизированы, выбор аппаратных средств обоснован, оценка составляет 18-20 баллов.

### 3. Шкала оценки

В общей оценке по дисциплине баллы за контрольную работу учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

Контрольная работа считается сданной, если средняя сумма баллов по всем заданиям составляет не менее 10 баллов (из 20 максимально возможных).

Общее количество баллов формируется из суммы баллов, полученных на лекционных занятиях (до 30 баллов), лабораторных работах (до 30 баллов), за выполнение контрольной работы (до 20 баллов) и баллы, полученные на зачете (максимум 20 баллов). Итоговая оценка в баллах по дисциплине составляется из суммы баллов, полученных за участие на лекционных занятиях (максимум 30 баллов), выполнение и защиту лабораторных работ (максимум 30 баллов), выполнение контрольной работы (максимум 20 баллов) и сдачу зачета (максимум 20 баллов) в письменной форме.

Диапазон баллов рейтинга	Оценка ECTS	Традиционная (4-уровневая) шкала оценки	
98-100	A+	отлично	зачтено
93-97	A		
90-92	A-		
87-89	B+		
83-86	B	хорошо	
80-82	B-		
77-79	C+		
73-76	C		
70-72	C-	удовл.	
67-69	D+		
63-66	D		
60-62	D-		
50-59	E	неуд.	незачтено
25-49	FX		
0-24	F		

### 4. Пример варианта контрольной работы

Разработка графической программы, имитирующей затухающие колебания маятника на веревке с использованием численного метода решения дифференциального уравнения.

Разработка графической программы, имитирующей затухающие колебания пружинного маятника с использованием численного метода решения дифференциального уравнения.

Разработка программы гармонического анализа периодической функции, задаваемой графически при помощи «мышки».

Разработка программы, выполняющей расчет определенного интеграла с задаваемыми подынтегральными функциями и диапазоном интегрирования.

Разработка экранной заставки «шарики», на которой шарики упруго соударяются со стенками и с самими собой, и угол соударения равен углу отражения.