

24

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГБОУ ВО «СГУ имени Н.Г. Чернышевского»

Факультет компьютерных наук и информационных технологий



УТВЕРЖДАЮ

OR

2016 г.

Рабочая программа дисциплины

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

Направление подготовки бакалавриата
38.03.05 – БИЗНЕС ИНФОРМАТИКА

Профиль подготовки бакалавриата
УПРАВЛЕНИЕ БИЗНЕС-ПРОЦЕССАМИ

Квалификация (степень) выпускника
Бакалавр

Форма обучения
Очная

Саратов,
2016 год

1. Цели освоения дисциплины

Целями освоения дисциплины «Информатика и программирование» являются формирование общих представлений об основных понятиях, методах и принципах разработки программных продуктов, развитие у студентов алгоритмического мышления, способностей к формализации, умение анализировать, находить и исправлять ошибки в собственном и чужом программном коде, развитие у студентов компетенций в области разработки и применения программных продуктов в профессиональной деятельности.

2. Место дисциплины в структуре ООП

Данная учебная дисциплина относится к вариативной части Блока 1 «Дисциплины (Модули)» ООП и направлена на формирование у обучающихся общепрофессиональных и профессиональных компетенций.

Для успешного усвоения данной дисциплины необходимы компетенции, сформированные у студентов в результате обучения в средней образовательной школе при изучении дисциплин, связанных с информатикой и программированием.

Компетенции, сформированные студентами при изучении данной дисциплины, используются при изучении следующих дисциплин «Операционные системы, среды и оболочки», «Информационные системы и технологии», «Основы объектно-ориентированного программирования», а также при написании курсовых и выпускных квалификационных работ.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины студент должен получить начальный уровень владения следующими компетенциями:

- способность к самоорганизации и самообразованию (ОК-7);
- способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности (ОПК-1);
- способность работать с компьютером как средством управления информацией, работать с информацией из различных источников, в том числе в глобальных компьютерных сетях (ОПК-3);
- умение разрабатывать контент и ИТ-сервисы предприятия и интернет-ресурсов (ПК-16);
- способность использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации по теме исследования (ПК-18);

В результате изучения указанных компетенций обучающийся должен:

Код компетенции	Планируемые результаты обучения
1	2
ОК-7	<p>Знать</p> <ul style="list-style-type: none"> – принципы организации рабочего времени при выполнении заданий; <p>Уметь</p> <ul style="list-style-type: none"> – работать самостоятельно, формулировать и представлять результаты своей деятельности; точно представлять знания по информатике в устной и письменной форме; организовывать свое рабочее время и пользоваться программами самообразования <p>Владеть</p> <ul style="list-style-type: none"> – способностью к самоорганизации и к самообразованию; навыками самостоятельной работы.
ОПК-1	<p>Знать</p> <ul style="list-style-type: none"> – информационно-коммуникационные технологии, применяемые для решения стандартных задач профессиональной деятельности – структуру программного обеспечения; – основные виды средств и сред разработки программного обеспечения и методы работы с ними; – алгоритмы обработки числовой и текстовой информации; – способы записи алгоритмов; – средства реализации алгоритма на языке программирования. <p>Уметь</p> <ul style="list-style-type: none"> – использовать информационно-коммуникационные технологии, информационные ресурсы и библиографические базы данных в решении профессиональных задач – работать в различных операционных системах и системных оболочках, использовать различные среды программирования – формулировать постановку задачи исходя из нужд заказчика, выбирать метод решения задачи; – создавать или выбирать алгоритм, и реализовывать его на языке программирования; – проводить отладку и тестирование программы. <p>Владеть</p> <ul style="list-style-type: none"> – способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры – основами алгоритмизации и программирования на языках высокого уровня при решении прикладных задач; – представлением о современных технических и программных средствах реализации информационных процессов.

1	2
ОПК-3	<p>Знать</p> <ul style="list-style-type: none"> – основные методы и подходы к поиску, сбору, обработке, анализу и систематизации информации, использованию компьютера и глобальных компьютерных сетей для подготовки обзоров, отчетов и научных публикаций <p>Уметь</p> <ul style="list-style-type: none"> – применять полученные в процессе обучения знания при решении задач профессиональной деятельности; самостоятельно проводить анализ результатов научно-исследовательской работы, делать обоснованные выводы <p>Владеть</p> <ul style="list-style-type: none"> – информационными компьютерными технологиями, необходимыми при выполнении научно-исследовательской работы
ПК-16	<p>Знать</p> <ul style="list-style-type: none"> – понятия информатики: данные, информация, знания, информационные процессы, информационные системы и технологии; – представление данных и основы алгоритмизации; – основные конструкции языков программирования; – абстрактные структуры данных, их представление в ЭВМ и способы использования при решении конкретных задач; – методы структурного программирования. <p>Уметь</p> <ul style="list-style-type: none"> – составлять алгоритмы решений для прикладных задач; – записывать алгоритмы на алгоритмических языках с использованием современных технологий программирования; – отлаживать и тестировать программные продукты. <p>Владеть</p> <ul style="list-style-type: none"> – навыками программирования в современных средах.
ПК-18	<p>Знать</p> <ul style="list-style-type: none"> – понятия информатики: данные, информация, знания, информационные процессы, информационные системы и технологии; – представление данных и основы алгоритмизации; – основные конструкции языков программирования; – абстрактные структуры данных, их представление в ЭВМ и способы использования при решении конкретных задач; – методы структурного программирования. <p>Уметь</p> <ul style="list-style-type: none"> – применять типовые подходы к разработке программного обеспечения. <p>Владеть</p> <ul style="list-style-type: none"> – навыками организации проектирования программного обеспечения, навыками оценки качественных и количественных характеристик программного обеспечения.

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 8 зач. ед., 288 часов

№ п/п	Раздел дисциплины	Се-местр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)						Формы текущего контроля успеваемости (по неделям семестра). Формы промежуточной аттестации (по семестрам)
				Всего часов	Лекции	Лабораторные занятия	Практические занятия	Самостоятельная работа	КСР	
1	2	3	4	5	6	7	8	9	10	11
1.	История развития информатики и программирования	1	1	16	2	2		12		Контрольная работа на 17 неделе
2.	Основные парадигмы программирования. Основные языки программирования	1	2	16	2	2		12		Контрольная работа на 17 неделе
3.	Основные конструкции языков программирования высокого уровня	1	3–6	28	8	8		12		Контрольная работа на 17 неделе
4.	Основные типы данных языков программирования высокого уровня на примере C++	1	7–10	28	8	8		12		Контрольная работа на 17 неделе
5.	Структурное программирование на примере языка программирования C++	1	11–13	22	6	6		10		Контрольная работа на 17 неделе
6.	Абстрактные типы данных. Списки	1	14–18	34	10	10		12	2	Контрольная работа на 17 неделе
Промежуточная аттестация										Экзамен
ИТОГО				180	36	36		70	2	36
7.	Абстрактные типы данных. Деревья и графы	2	1–4	18	8	4		6		Контрольная работа на 15 неделе

1	2	3	4	5	6	7	8	9	10	11
8.	Понятие алгоритмов. Алгоритмы сортировки и поиска	2	5–8	18	8	4		6		Контрольная работа на 15 неделе
9.	Обобщенное программирование на примере языка C++. Стандартная библиотека шаблонов	2	9–12	18	8	4		6		Контрольная работа на 15 неделе
10.	Разработка, тестирование, документирование программного продукта	2	13–16	18	8	4		4	2	Контрольная работа на 15 неделе
Промежуточная аттестация										Экзамен
ИТОГО				108	32	16		22	2	36
ВСЕГО				288	68	52		92	4	72

ИСТОРИЯ РАЗВИТИЯ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

Понятие информатики. Понятие информационных технологий. Виды информационных технологий. История развития информатики. Теории и факты развития информатики. Современные концепции и принципы, связанные с информатикой, как с наукой.

Ученые, связанные с развитием информационных технологий. Современное состояние информационных технологий и программирования. Стандарты в области информационных технологий и программирования.

ОСНОВНЫЕ ПАРАДИГМЫ ПРОГРАММИРОВАНИЯ. ОСНОВНЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Понятие парадигмы программирования. Современные парадигмы программирования. Их достоинства и недостатки.

Языки программирования низкого уровня. Языки программирования высокого уровня. История развития языков программирования.

ОСНОВНЫЕ КОНСТРУКЦИИ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ ВЫСОКОГО УРОВНЯ

Язык программирования C++. Директивы препроцессора. Сборка и компиляция проекта.

Основные конструкции: алфавит, ключевые слова, идентификаторы, блоки. Операции, приоритет операций.

Операторы. Простой и составной оператор. Операторы присвоения. Операторы ветвления. Операторы цикла. Операторы управления.

Переменная. Время жизни переменной, область видимости переменной.

ОСНОВНЫЕ ТИПЫ ДАННЫХ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ ВЫСОКОГО УРОВНЯ НА ПРИМЕРЕ C++

Понятие типа данных. Базовые типы данных: целочисленные, логические, вещественные, символьные. Приведение типов данных.

Составные типы данных: массивы, строки, перечисления, объединения, структуры, указатели.

Понятие потока. Потоки ввода-вывода, файловые потоки, строковые потоки.

СТРУКТУРНОЕ ПРОГРАММИРОВАНИЕ НА ПРИМЕРЕ ЯЗЫКА ПРОГРАММИРОВАНИЯ C++

Понятие функции в языке C++. Функции, возвращающие и не возвращающие значение. Глобальные и локальные переменные. Формальные и фактические переменные. Способы передачи параметров в функцию: по значению, по указателю и по ссылке.

Понятие рекурсии. Математические рекурсивные функции. Примеры рекурсивных функций.

Перегрузка функций. Шаблоны функций.

Исключения. Виды исключений.

АБСТРАКТНЫЕ ТИПЫ ДАННЫХ. СПИСКИ

Понятие абстрактных типов данных. Примеры абстрактных типов данных. Способы представления абстрактных типов данных.

Абстрактный тип данных — список. Виды списков. Стек, очередь, односвязный и двусвязный список, кольцевой список.

АБСТРАКТНЫЕ ТИПЫ ДАННЫХ. ДЕРЕВЬЯ И ГРАФЫ

Абстрактный тип данных — деревья. Виды деревьев. Обходы деревьев. Использование деревьев для алгоритмов поиска и сортировки.

Абстрактный тип данных — граф. Виды графов. Способы представления графов. Обходы графов.

ПОНЯТИЕ АЛГОРИТМОВ. АЛГОРИТМЫ СОРТИРОВКИ И ПОИСКА

Понятие алгоритма. Свойства алгоритмов. Виды алгоритмов.

Сложность алгоритмов. Временная сложность алгоритмов. Оценка временной сложности.

Методы разработки алгоритмов. Алгоритмы сортировки и поиска.

Элементарные алгоритмы сортировки. Сортировки по методу декомпозиции. Сортировки за линейное время.

Алгоритмы поиска в строке.

ОБОБЩЕННОЕ ПРОГРАММИРОВАНИЕ НА ПРИМЕРЕ ЯЗЫКА C++. СТАНДАРТНАЯ БИБЛИОТЕКА ШАБЛОНОВ

Парадигма обобщенного программирования. Стандартная библиотека шаблонов.

Контейнеры. Примеры контейнеров, их преимущества и недостатки.

Итераторы. Виды итераторов.

Алгоритмы. Виды алгоритмов. Применение алгоритмов для различных контейнеров.

РАЗРАБОТКА, ТЕСТИРОВАНИЕ, ДОКУМЕНТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА

Постановка задачи. Формализация предметной области программного проекта. Построение модели. Выбор языка программирования, разработка спецификации для компонентов программного продукта.

Проектирование структуры программы. Отладка программы. Чтение, понимание и выделение главной идеи прочитанного исходного кода

Кодирование и документирование программы. Единая система программной документации. Взаимодействие с другими пользователями и разработчиками.

План лабораторных занятий

На лабораторных занятиях студенты разрабатывают алгоритмы поставленных заданий, реализуют разработанные алгоритмы на языке программирования C++, проверяют корректность работы реализованных программ и объясняют ход решения преподавателю.

Задания для лабораторного практикума приведены в ФОС, раздел «Задания для лабораторных занятий».

№ занятия	Тема	Задания для лабораторного практикума
1–4	Основные конструкции языков программирования высокого уровня	I.1–I.20, II.1–II.20, III.1–III.20, IV.1–IV.20
5–9	Основные типы данных языков программирования высокого уровня на примере C++	V.1–V.20, VI.1–VI.20, VII.1–VII.20, VIII.1–VIII.20, IX.1–IX.20
10–14	Структурное программирование на примере языка программирования C++	X.1–X.20, XI.1–XI.20
15–18	Абстрактные типы данных. Списки	XII.1–XII.20, XIII.1–XIII.20
19–20	Абстрактные типы данных. Деревья и графы	XIV.1–XIV.20, XV.1–XV.20

21–23	Понятие алгоритмов. Алгоритмы сортировки и поиска	XVI.1–XVI.20, XVII.1–XVII.20
24–25	Обобщенное программирование на примере языка C++. Стандартная библиотека шаблонов	XVIII.1–XVIII.20
26	Разработка, тестирование, документирование программного продукта	XIX.1

5. Образовательные технологии

Предусматривается широкое использование в учебном процессе активных и интерактивных форм проведения занятий: дистанционные технологии, мультимедийные презентации теоретического материала.

При обучении лиц с ограниченными возможностями здоровья и инвалидов используются подходы, способствующие созданию безбарьерной образовательной среды: технологии дифференциации и индивидуализации обучения, увеличивается время на самостоятельное освоение материала. Используется сочетание разных форм и способов передачи учебной информации: вербальный, невербальный, с использованием средств визуализации информации (презентации) и разных способов отчетности (письменно, устно, с использованием электронных дистанционных технологий).

6. Учебно-методическое обеспечение самостоятельной работы студентов. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

В рамках самостоятельной работы студенты самостоятельно прорабатывают практические примеры для наилучшего усвоения методов и алгоритмов теоретического курса, готовят реферат по заданным темам, приведенным в ФОС

Фонд оценочных средств дисциплины включает в себя задания для самостоятельной работы (методические рекомендации, критерии оценивания и список заданий), задания для практических занятий (методические рекомендации, критерии оценивания и список заданий), задания для лабораторных занятий (методические рекомендации, критерии оценивания и список заданий) по соответствующим разделам, задания для контрольной работы (методические рекомендации, критерии оценивания и варианты заданий), контрольные вопросы, вопросы для проведения промежуточной аттестации.

7. Данные для учета успеваемости студентов в БАРС

Таблица максимальных баллов по видам учебной деятельности

1	2	3	4	5	6	7	8	9
Се- местр	Лекции	Лабораторные занятия	Практические занятия	Самостоятельная работа	Автоматизированное тестирование	Другие виды учебной деятельности	Промежуточная аттестация	Итого
1	5	35	0	10	0	10	40	100
2	5	35	0	10	0	10	40	100

Программа оценивания учебной деятельности студента Семестр 1

Лекции

Посещаемость, опрос, активность и др. за один семестр — от 0 до 5 баллов.

Лабораторные занятия

Контроль выполнения лабораторных заданий в течение одного семестра — от 0 до 35 баллов.

Практические занятия

Не предусмотрены.

Самостоятельная работа

Выполнение заданий для самостоятельной работы — от 0 до 10 баллов.

Автоматизированное тестирование

Не предусмотрено.

Другие виды учебной деятельности

Контрольная работа №1 (от 0 до 5 баллов).

Таким образом в течение семестра студент может получить от 0 до 10 баллов.

Промежуточная аттестация

Методика оценивания знаний обучающихся по дисциплине «Информатика и программирование» в ходе промежуточной аттестации

25–40 баллов — ответ на «отлично»:

Ответ студента содержит:

- глубокое знание программного материала, а также основного содержания и новаций лекционного курса по сравнению с учебной литературой;
- знание концептуально-понятийного аппарата всего курса;
- знание монографической литературы по курсу,
- также свидетельствует о способности:
- самостоятельно критически оценивать основные положения курса;

– увязывать теорию с практикой.

15–24 баллов — ответ на «хорошо»:

Ответ студента свидетельствует:

- о полном знании материала по программе;
- о знании рекомендованной литературы,
- а также содержит в целом правильное, но не всегда точное и аргументированное изложение материала.

10–14 баллов — ответ на «удовлетворительно»:

Ответ студента содержит:

- поверхностные знания важнейших разделов программы и содержания лекционного курса;
- затруднения с использованием научно-понятийного аппарата и терминологии курса;
- стремление логически четко построить ответ, а также свидетельствует о возможности последующего обучения.

0–9 баллов — неудовлетворительный ответ :

Студент имеет существенные пробелы в знании основного материала по программе и допускает принципиальные ошибки при изложении материала.

Таким образом, максимально возможная сумма баллов за все виды учебной деятельности студента за 1-й семестр по дисциплине «Информатика и программирование» составляет 100 баллов.

Таблица пересчета полученной студентом суммы баллов по дисциплине «Информатика и программирование» в оценку (экзамен):

меньше 61 баллов	«неудовлетворительно»
от 61 до 74 баллов	«удовлетворительно»
от 75 до 90 баллов	«хорошо»
от 91 баллов и более	«отлично»

Семестр 2

Лекции

Посещаемость, опрос, активность и др. за один семестр — от 0 до 5 баллов.

Лабораторные занятия

Контроль выполнения лабораторных заданий в течение одного семестра — от 0 до 35 баллов.

Практические занятия

Не предусмотрены

Самостоятельная работа

Выполнение заданий для самостоятельной работы — от 0 до 10 баллов.

Автоматизированное тестирование

Не предусмотрено.

Другие виды учебной деятельности

Контрольная работа (от 0 до 5 баллов).

Таким образом в течение семестра студент может получить от 0 до 10 баллов.

Промежуточная аттестация

Методика оценивания знаний обучающихся по дисциплине «Информатика и программирование» в ходе промежуточной аттестации

25–40 баллов — ответ на «отлично»:

Ответ студента содержит:

- глубокое знание программного материала, а также основного содержания и новаций лекционного курса по сравнению с учебной литературой;
- знание концептуально-понятийного аппарата всего курса;
- знание монографической литературы по курсу,
- также свидетельствует о способности:
- самостоятельно критически оценивать основные положения курса;
- увязывать теорию с практикой.

15–24 баллов — ответ на «хорошо»:

Ответ студента свидетельствует:

- о полном знании материала по программе;
- о знании рекомендованной литературы,
- а также содержит в целом правильное, но не всегда точное и аргументированное изложение материала.

10–14 баллов — ответ на «удовлетворительно»:

Ответ студента содержит:

- поверхностные знания важнейших разделов программы и содержания лекционного курса;
- затруднения с использованием научно-понятийного аппарата и терминологии курса;
- стремление логически четко построить ответ, а также свидетельствует о возможности последующего обучения.

0–9 баллов — неудовлетворительный ответ :

Студент имеет существенные пробелы в знании основного материала по программе и допускает принципиальные ошибки при изложении материала.

Таким образом, максимально возможная сумма баллов за все виды учебной деятельности студента за 2-й семестр по дисциплине «Информатика и программирование» составляет 100 баллов.

Таблица пересчета полученной студентом суммы баллов

по дисциплине «Информатика и программирование» в оценку (экзамен):

меньше 61 баллов	«неудовлетворительно»
от 61 до 74 баллов	«удовлетворительно»
от 75 до 90 баллов	«хорошо»
от 91 баллов и более	«отлично»

8. Учебно-методическое и информационное обеспечение дисциплины «Информатика и программирование»

а) основная литература:

1. Кузин А.В. Программирование на языке Си, Москва : Издательство "Инфра-М 2015. [Электронный ресурс] <http://znanium.com/go.php?id=505194> ✓

б) дополнительная литература:

2. Немцова Т. И. Программирование на языке высокого уровня. Программирование на языке C++. Учебное пособие Москва : Издательство "Инфра-М 2012. <http://znanium.com/go.php?id=244875> ✓
3. Лафоре Р. Объектно-ориентированное программирование в C++. Классика Computer Science, СПб., 2011. [Электронный ресурс] <http://ibooks.ru/reading.php?short=1&isbn=978-5-4237-0038-6> ✓
Ладья-

в) Интернет-ресурсы:

4. Справочник по C++, [Электронный ресурс] <http://msdn.microsoft.com/ru-ru/library/3bstk3k5.aspx/>

г) программное обеспечение:

5. Свободное программное обеспечение: Microsoft Visual Studio Express, Code::Blocks IDE, Eclipse +CDT, NetBeans.

9. Материально-техническое обеспечение учебной дисциплины «Информатика и программирование»

Для проведения лекционных и практических занятий необходима доска. Для проведения лабораторных занятий необходимо наличие компьютерного класса с установленным программным обеспечением и доступом к сети Интернет.

Программа составлена в соответствии с требованиями ФГОС ВО для направления 38.03.05 «БИЗНЕС ИНФОРМАТИКА».

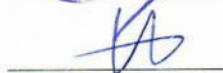
Авторы

Зав. кафедрой технологий
программирования, к. ф.-м. н.



Батраева И. А.

доцент, к. ф.-м. н.



Иванова А. С.

Программа одобрена на заседании кафедры математической кибернетики и компьютерных наук от «30» сентября 2016 г., протокол № 1.

Зав. кафедрой математической
кибернетики и компьютерных
наук, к. ф.-м. н.



Миронов С. В.

Декан факультета КНиИТ,
к. ф.-м. н., доцент



Федорова А. Г.

Декан
механико-математического
факультета, к. ф.-м. н., доцент



Захаров А. М.