

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
"Казанский (Приволжский) федеральный университет"  
Институт физики



УТВЕРЖДАЮ  
Проректор по образовательной деятельности КФУ  
Проф. Д.А. Таюрский

\_\_\_\_\_» \_\_\_\_\_ 20\_\_ г.

подписано электронно-цифровой подписью

### Программа дисциплины

Технологии и методы программирования Б1.Б.20

Направление подготовки: 10.03.01 - Информационная безопасность

Профиль подготовки: Безопасность автоматизированных систем

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

**Автор(ы):**

Андрианова А.А., Васильев А.В.

**Рецензент(ы):**

Тагиров Р.Р.

**СОГЛАСОВАНО:**

Заведующий(ая) кафедрой: Латыпов Р. Х.

Протокол заседания кафедры No \_\_\_\_\_ от "\_\_\_\_\_" \_\_\_\_\_ 201\_\_ г

Учебно-методическая комиссия Института физики:

Протокол заседания УМК No \_\_\_\_\_ от "\_\_\_\_\_" \_\_\_\_\_ 201\_\_ г

Регистрационный No 6127118

Казань  
2018

## Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре основной образовательной программы
3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля
4. Структура и содержание дисциплины/ модуля
5. Образовательные технологии, включая интерактивные формы обучения
6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов
7. Литература
8. Интернет-ресурсы
9. Материально-техническое обеспечение дисциплины/модуля согласно утвержденному учебному плану

Программу дисциплины разработал(а)(и) доцент, к.н. Андрианова А.А. кафедра системного анализа и информационных технологий отделение фундаментальной информатики и информационных технологий , Anastasiya.Andrianova@kpfu.ru ; заместитель директора по научной деятельности Васильев А.В. Директорат Института ВМ и ИТ Институт вычислительной математики и информационных технологий , Alexander.Vasiliev@kpfu.ru

### 1. Цели освоения дисциплины

Данная дисциплина предполагает изучение одной из современных технологий программирования - объектно-ориентированной технологии, на основе которой строится большинство современных информационных систем. Данная технология изучается на примере языка программирования C#.

Практические занятия по курсу производятся с помощью среды разработки Visual Studio 2010.

### 2. Место дисциплины в структуре основной образовательной программы высшего профессионального образования

Данная учебная дисциплина включена в раздел " Б1.Б.20 Дисциплины (модули)" основной образовательной программы 10.03.01 Информационная безопасность и относится к базовой (общепрофессиональной) части. Осваивается на 2 курсе, 3, 4 семестры.

Дисциплина "Технологии и методы программирования" изучается на 2 курсе в 3 и 4 семестрах обучения бакалавров. Предварительно студенты должны получить базовые знания и навыки по программированию при изучении дисциплины "Информатика" и "Языки программирования". В дальнейшем знания по данной дисциплине могут использоваться в других дисциплинах профессионального цикла, при написании курсовых и выпускной работы бакалавра.

### 3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля

В результате освоения дисциплины формируются следующие компетенции:

| Шифр компетенции                        | Расшифровка приобретаемой компетенции   |
|---|---|
| ОПК-4<br>(профессиональные компетенции) | способностью понимать значение информации в развитии современного общества, применять информационные технологии для поиска и обработки информации |
| ПК-3<br>(профессиональные компетенции)  | способностью администрировать подсистемы информационной безопасности объекта защиты   |

В результате освоения дисциплины студент:

1. должен знать:

- основные принципы объектно-ориентированного программирования.
- основные приемы программирования на языке C#.

2. должен уметь:

- создавать собственные классы на языке программирования C#, а также использовать классы из библиотек этого языка;

- создавать сложные приложения с использованием расширенного пользовательского интерфейса и привлечением внешних источников данных.

3. должен владеть:

- навыками разработки консольных приложений в стиле объектно-ориентированного программирования и Windows-приложений на языке программирования C#.

4. должен демонстрировать способность и готовность:

- применять полученные знания и навыки в дальнейшем обучении, а также в своей последующей профессиональной деятельности.

#### 4. Структура и содержание дисциплины/ модуля

Общая трудоемкость дисциплины составляет 5 зачетных(ые) единиц(ы) 180 часа(ов).

Форма промежуточного контроля дисциплины зачет в 3 семестре; экзамен в 4 семестре.

Суммарно по дисциплине можно получить 100 баллов, из них текущая работа оценивается в 50 баллов, итоговая форма контроля - в 50 баллов. Минимальное количество для допуска к зачету 28 баллов.

86 баллов и более - "отлично" (отл.);

71-85 баллов - "хорошо" (хор.);

55-70 баллов - "удовлетворительно" (удов.);

54 балла и менее - "неудовлетворительно" (неуд.).

#### 4.1 Структура и содержание аудиторной работы по дисциплине/ модулю

##### Тематический план дисциплины/модуля

| N  | Раздел<br>Дисциплины/<br>Модуля          | Семестр | Неделя<br>семестра | Виды и часы<br>аудиторной работы,<br>их трудоемкость<br>(в часах) |                         |                        | Текущие формы<br>контроля         |
|----|--|---------|--------------------|---|-------------------------|------------------------|-----------------------------------|
|    |  |         |                    | Лекции  | Практические<br>занятия | Лабораторные<br>работы |                                   |
| 1. | Тема 1. Понятие<br>класса. Инкапсуляция. | 3       | 0                  | 3   | 0                       | 6                      | Письменное<br>домашнее<br>задание |
| 2. | Тема 2. Конструкторы<br>и деструкторы.   | 3       | 0                  | 1   | 0                       | 6                      | Письменное<br>домашнее<br>задание |
| 3. | Тема 3. Свойства<br>классов.             | 3       | 0                  | 2   | 0                       | 6                      | Письменное<br>домашнее<br>задание |
| 4. | Тема 4. Перегрузка<br>операций в классе. | 3       | 0                  | 2   | 0                       | 6                      | Письменное<br>домашнее<br>задание |

| N   | Раздел Дисциплины/ Модуля                             | Семестр | Неделя семестра | Виды и часы аудиторной работы, их трудоемкость (в часах) |                      |                     | Текущие формы контроля                            |
|-----|---|---------|-----------------|--|----------------------|---------------------|---|
|     |   |         |                 | Лекции   | Практические занятия | Лабораторные работы |   |
| 5.  | Тема 5. Наследование.                                 | 3       | 0               | 6  | 0                    | 6                   | Письменное домашнее задание                       |
| 6.  | Тема 6. Полиморфизм и виртуальные функции.            | 3       | 0               | 4  | 0                    | 6                   | Контрольная работа<br>Письменное домашнее задание |
| 7.  | Тема 7. Обработка исключительных ситуаций.            | 4       | 0               | 2  | 0                    | 0                   | Письменное домашнее задание                       |
| 8.  | Тема 8. Объектно-ориентированные особенности языка C# | 4       | 0               | 2  | 0                    | 0                   | Письменное домашнее задание                       |
| 9.  | Тема 9. Делегаты и события                            | 4       | 0               | 4  | 0                    | 18                  | Письменное домашнее задание                       |
| 10. | Тема 10. Работа с коллекциями                         | 4       |                 | 2  | 0                    | 0                   | Письменное домашнее задание                       |
| 11. | Тема 11. Работа с файлами                             | 4       |                 | 4  | 0                    | 18                  | Письменное домашнее задание                       |
| 12. | Тема 12. Обобщенные классы (шаблоны).                 | 4       |                 | 4  | 0                    | 0                   | Контрольная работа<br>Письменное домашнее задание |
|     | Тема . Итоговая форма контроля                        | 3       |                 | 0  | 0                    | 0                   | Зачет   |
|     | Тема . Итоговая форма контроля                        | 4       |                 | 0  | 0                    | 0                   | Экзамен   |
|     | Итого   |         |                 | 36   | 0                    | 72                  |   |

## 4.2 Содержание дисциплины

### Тема 1. Понятие класса. Инкапсуляция.

#### *лекционное занятие (3 часа(ов)):*

Инкапсуляция как принцип объектно-ориентированного программирования. Класс как реализация принципа инкапсуляции. Члены класса - переменные (данные-члены) и методы (функции-члены) класса. Управление доступом к членам класса. Операции доступа к членам класса.

**лабораторная работа (6 часа(ов)):**

Написание программ с разработкой классов, например, класса рациональная дробь.

**Тема 2. Конструкторы и деструкторы.**

**лекционное занятие (1 часа(ов)):**

Конструкторы и деструкторы. Назначение конструкторов и деструкторов. Отличия конструкторов и деструкторов от обычных функций-членов класса. Конструкторы по умолчанию. Необходимость перегрузки конструкторов.

**лабораторная работа (6 часа(ов)):**

Написание программ с разработкой классов, например, класса матрица.

**Тема 3. Свойства классов.**

**лекционное занятие (2 часа(ов)):**

Свойства классов. Внедрение объектов. Особенности вызова конструкторов внедренных объектов. Вложенные классы.

**лабораторная работа (6 часа(ов)):**

Написание программ с разработкой классов, например, класса односвязный список.

**Тема 4. Перегрузка операций в классе.**

**лекционное занятие (2 часа(ов)):**

Перегрузка операций в классе. Общие правила переопределения операций. Перегрузка бинарных операций. Перегрузка унарных операций. Особенности перегрузки префиксной и постфиксной форм операции инкремента.

**лабораторная работа (6 часа(ов)):**

Написание программ с разработкой классов, например, добавление методов арифметических операций для класса матрица.

**Тема 5. Наследование.**

**лекционное занятие (6 часа(ов)):**

Наследование. Понятие наследования. Базовый и производный классы. Порядок вызова конструкторов и деструкторов при наследовании. Управление доступом при наследовании. Спецификатор доступа protected.

**лабораторная работа (6 часа(ов)):**

Определение в классе квадратная матрица новых методов получения обратной матрицы и определителя.

**Тема 6. Полиморфизм и виртуальные функции.**

**лекционное занятие (4 часа(ов)):**

Полиморфизм и виртуальные функции. Особенности использования указателей на объект базового класса. Виртуальные функции как основной способ реализации полиморфизма. Основные примеры использования виртуальных функций. Абстрактные методы и классы, интерфейсы.

**лабораторная работа (6 часа(ов)):**

Создание интерфейса для формирования рисунка в виде списка фигур.

**Тема 7. Обработка исключительных ситуаций.**

**лекционное занятие (2 часа(ов)):**

Обработка исключительных ситуаций. Возможности программной обработки исключительных ситуаций. Операторы try, catch, throw. Создание иерархии исключений.

**Тема 8. Объектно-ориентированные особенности языка C#**

**лекционное занятие (2 часа(ов)):**

Объектно-ориентированные особенности языка C#: Свойства. Индексаторы. Итераторы. Статические члены класса.

**Тема 9. Делегаты и события**

**лекционное занятие (4 часа(ов)):**

Делегаты и события Определение простого и комбинированного делегата. Применение делегатов. Использование делегатов для реализации событий. Примеры использования событий.

**лабораторная работа (18 часа(ов)):**

Создание оконных приложений с обработкой событий.

**Тема 10. Работа с коллекциями**

**лекционное занятие (2 часа(ов)):**

Работа с коллекциями Классы из пространства имен Collections. Коллекции-контейнеры - динамические массивы, списки, словари. Использование нумераторов. Сравнение элементов коллекций. Коллекции-шаблоны.

**Тема 11. Работа с файлами**

**лекционное занятие (4 часа(ов)):**

Работа с файлами Потоки ввода. Потоки вывода. Текстовые и двоичные файлы.

**лабораторная работа (18 часа(ов)):**

Решение задач на использование файлов - чтение исходных данных и сохранение результатов в файлы.

**Тема 12. Обобщенные классы (шаблоны).**

**лекционное занятие (4 часа(ов)):**

Обобщенные классы (шаблоны). Понятие шаблона класса. Использование шаблона класса. Ограничения при использовании шаблонов.

**4.3 Структура и содержание самостоятельной работы дисциплины (модуля)**

| N  | Раздел Дисциплины                          | Семестр | Неделя семестра | Виды самостоятельной работы студентов | Трудоемкость (в часах) | Формы контроля самостоятельной работы |
|----|--|---------|-----------------|---------------------------------------|------------------------|---------------------------------------|
| 1. | Тема 1. Понятие класса. Инкапсуляция.      | 3       | 0               | подготовка домашнего задания          | 2                      | домашнее задание                      |
| 2. | Тема 2. Конструкторы и деструкторы.        | 3       | 0               | подготовка домашнего задания          | 4                      | домашнее задание                      |
| 3. | Тема 3. Свойства классов.                  | 3       | 0               | подготовка домашнего задания          | 4                      | домашнее задание                      |
| 4. | Тема 4. Перегрузка операций в классе.      | 3       | 0               | подготовка домашнего задания          | 2                      | домашнее задание                      |
| 5. | Тема 5. Наследование.                      | 3       | 0               | подготовка домашнего задания          | 4                      | домашнее задание                      |
| 6. | Тема 6. Полиморфизм и виртуальные функции. | 3       | 0               | подготовка домашнего задания          | 1                      | домашнее задание                      |
|    |  |         |                 | подготовка к контрольной работе       | 1                      | контрольная работа                    |
| 7. | Тема 7. Обработка исключительных ситуаций. | 4       | 0               | подготовка домашнего задания          | 4                      | домашнее задание                      |



| №     | Раздел Дисциплины                                     | Семестр | Неделя семестра | Виды самостоятельной работы студентов | Трудоемкость (в часах) | Формы контроля самостоятельной работы |
|-------|---|---------|-----------------|---------------------------------------|------------------------|---------------------------------------|
| 8.    | Тема 8. Объектно-ориентированные особенности языка C# | 4       | 0               | подготовка домашнего задания          | 4                      | домашнее задание                      |
| 9.    | Тема 9. Делегаты и события                            | 4       | 0               | подготовка домашнего задания          | 4                      | домашнее задание                      |
| 10.   | Тема 10. Работа с коллекциями                         | 4       |                 | подготовка домашнего задания          | 2                      | домашнее задание                      |
| 11.   | Тема 11. Работа с файлами                             | 4       |                 | подготовка домашнего задания          | 2                      | домашнее задание                      |
| 12.   | Тема 12. Обобщенные классы (шаблоны).                 | 4       |                 | подготовка домашнего задания          | 1                      | домашнее задание                      |
|       |   |         |                 | подготовка к контрольной работе       | 1                      | контрольная работа                    |
| Итого |   |         |                 |                                       | 36                     |                                       |

## 5. Образовательные технологии, включая интерактивные формы обучения

Дисциплина представляет собой цикл лекционных и лабораторных (практических) занятий. Практические занятия посвящены выработке навыков разработки программ в объектно-ориентированном стиле на языке программирования C#. Практические занятия проходят в компьютерных классах с использованием интерактивной доски для наглядного представления алгоритмов и разработки программ на всех этапах ее создания и компиляции. Практические занятия проходят в интерактивной форме обсуждения решения различных задач или в активной форме самостоятельного решения задач студентами. Контроль за выполнением самостоятельной работы проявляется в функциональном тестировании выполненных студентами заданий на примерах, предложенных преподавателем.

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

### Тема 1. Понятие класса. Инкапсуляция.

домашнее задание, примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. Создать класс "Вектор на плоскости". Определить функции для получения суммы, разности скалярного произведения двух векторов, получения длины вектора, проверки ортогональности двух векторов и пр.

### Тема 2. Конструкторы и деструкторы.

домашнее задание, примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. В класс "Вектор на плоскости" добавить конструкторы, позволяющие инициализировать вектор разными способами.

### Тема 3. Свойства классов.

домашнее задание, примерные вопросы:



Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. В класс "Вектор на плоскости" добавит свойства и индексы для доступа к данным класса.

#### **Тема 4. Перегрузка операций в классе.**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. Добавить в класс "Вектор на плоскости" переопределенные операции суммы, скалярного произведения, сравнения двух векторов.

#### **Тема 5. Наследование.**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. Создать класс "Треугольник", определенный координатами трех вершин, и наследовать от него класс "Четырехугольник".

#### **Тема 6. Полиморфизм и виртуальные функции.**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. В классе "Треугольник" определить виртуальную функцию рисования фигуры и переопределить ее в классе "Четырехугольник".

контрольная работа , примерные вопросы:

Примерные варианты заданий для контрольной работы. 1. Постройте DLL с классом `Sorting`, содержащим различные методы сортировки. Методы должны быть универсальными и позволять сортировать объекты разных типов - персоны, машины, числа, строки. Методы должны быть функциями высших порядков, допускающими сортировать данные типа `T` по разным критериям, например, сортировать машины по маркам, по номерам, по фамилиям владельцев. В Windows-проекте постройте классы `Person`, `Car` и другие классы, демонстрирующие работу с классом `Sorting`. В клиентском классе предусмотрите построение набора функций, позволяющих сравнивать персон по разным критериям. Предусмотрите возможность задания таких функций лямбда-выражениями. В интерфейсе проекта предусмотрите возможность сравнения методов сортировки по времени. Постройте метод, вычисляющий время сортировки, как функцию высшего порядка, которой передается в качестве аргумента метод сортировки. 2. Постройте DLL, включающую класс `BinaryTreeSearch<K, T>` - бинарное дерево поиска. Элементы, хранимые в дереве, обладают ключом типа `K` и информационным полем типа `T`. Ключи элементов уникальны. Для дерева поиска справедливо свойство: Ключ элемента, хранимого в корне дерева, больше ключей всех элементов, хранимых в левом поддереве, и меньше ключей всех элементов, хранимых в правом поддереве. Постройте Windows-проект для работы с этим классом. 3. Постройте DLL с классами, реализующими сбалансированные бинарные деревья поиска - идеально сбалансированные деревья, АВЛ - деревья, красно-черные деревья. Постройте Windows-проект для работы с этим классом. Проект должен позволять оценить время работы на одних и тех же данных при использовании разных типов деревьев. 4. Постройте DLL с классами `Student` и `Teacher`, связанными взаимными отношениями клиент - поставщик. Обеспечьте возможность сериализации данных, используя атрибут сериализации и интерфейс `ISerializable`. 5. Напишите реализацию класса `DEQ (Double Ended Queue)`, задающего динамическую структуру данных - двустороннюю очередь, где добавление и удаление элементов выполняется на обоих концах очереди. Постройте семейство классов, начиная с абстрактного класса и заканчивая разными классами потомками. Постройте реализации, основанные на массивах и на списковой структуре данных. Постройте классы потомки, хранящие в очереди элементы фиксированного типа. Постройте Windows-проект, демонстрирующий полиморфизм построенного семейства классов. 6. Напишите реализацию класса `ListWithCursor`, задающего динамическую структуру данных - список с курсором. Курсор в этой структуре данных задает текущий элемент списка. В списке должны быть определены операции по перемещению курсора. Операция поиска элемента списка устанавливает курсор на найденном элементе списка. Операции добавления и удаления элементов списка выполняются по отношению элемента, на который указывает курсор. Постройте семейство классов, начиная с абстрактного класса и заканчивая разными классами потомками. Постройте реализации, основанные на массивах и на списковой структуре данных. Постройте классы потомки, хранящие в списке элементы фиксированного типа. Постройте Windows-проект, демонстрирующий полиморфизм построенного семейства классов.

### **Тема 7. Обработка исключительных ситуаций.**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. Обработка исключений при операциях с классом "Матрица".

### **Тема 8. Объектно-ориентированные особенности языка C#**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. Создание индексаторов, итераторов и свойств для класса "Квадратная матрица".

### **Тема 9. Делегаты и события**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. Создание класса для решения уравнений. Уравнение и метод решения задавать с помощью делегатов. Использование событий при создании оконных приложений.

### **Тема 10. Работа с коллекциями**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Типовой пример. Создание класса "Разреженная матрица" с использованием линейных списков. Создание класса "Граф" с помощью словарей.

### **Тема 11. Работа с файлами**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Организация файлового ввода-вывода в уже созданных классах (например, в классе "Разреженная матрица").

### **Тема 12. Обобщенные классы (шаблоны).**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Совместное и самостоятельное решение задач. Создание обобщенного класса "Множество", определение обобщенных операций с множествами.

контрольная работа , примерные вопросы:

Типовые варианты заданий для контрольной работы: 1. Создайте Windows-проект с классом Functions. Класс должен выполнять различные операции над функциями, принадлежащими функциональному типу, заданному делегатом. В число операций класса включите такие методы, как нахождение максимума, минимума, корней функции на заданном интервале, вычисление значения производной функции в заданной точке. В классе-клиенте, использующем сервисы класса Functions, используйте анонимные методы и лямбда-выражения. В интерфейсе проекта предусмотрите возможность задания функции как выражением, так и выбором из фиксированного списка. 2. Создайте Windows-проект с классами Person и Sorting. В классе Sorting предусмотрите разные методы сортировки, заданные функциями высших порядков. Аргументом метода сортировки должна быть функция, позволяющая сравнивать персоны. В клиентском классе предусмотрите построение набора функций, позволяющих сравнивать персоны по разным критериям. Предусмотрите возможность задания таких функций лямбда-выражениями. В интерфейсе проекта предусмотрите возможность сравнения методов сортировки по времени. Постройте метод, вычисляющий время сортировки, как функцию высшего порядка, которой передается в качестве аргумента метод сортировки. 3. Напишите реализацию класса ChessBoard, описывающий шахматную доску. Постройте семейство классов, начиная с абстрактного класса Figure с абстрактным методом Move и заканчивая разными классами потомками для конкретных фигур. Добавьте фигурам событие Moved, а в классе шахматной доски напишите обработчик этого события, разрешающий или запрещающий перемещение фигуры. Постройте Windows-проект, демонстрирующий работу построенного семейства классов.

### **Итоговая форма контроля**

зачет и экзамен

### **Итоговая форма контроля**

зачет и экзамен

Примерные вопросы к :

Вопросы к зачету

1. Понятие класса. Инкапсуляция.
2. Конструкторы и деструкторы.
3. Свойства классов.
4. Перегрузка операций в классе.
5. Наследование.
6. Полиморфизм и виртуальные функции.
7. Обработка исключительных ситуаций.
8. Объектно-ориентированные особенности языка C#
9. Делегаты и события
10. Работа с коллекциями

11. Работа с файлами

12. Обобщенные классы (шаблоны).

Билеты к экзамену по курсу "Технологии и методы программирования"

БИЛЕТ 1

1. Структура программы и основные конструкции

2. Написать программу для вычисления синуса с заданной точностью для заданного  $x$  по формуле

3. Написать программу, которая в текстовом файле заменит все вхождения одного заданного слова на другое заданное слово

БИЛЕТ 2

1. Операторы цикла

2. Написать программу для вычисления косинуса с заданной точностью для заданного  $x$  по формуле

3. Написать программу для вычисления коэффициентов  $p_0, p_1, p_2, \dots, p_n$  многочлена степени  $n$  по заданным корням

$x_1, x_2, \dots, x_n$

БИЛЕТ 3

1. Простейший ввод-вывод (клавиатура + экран)

2. Написать программу для вычисления экспоненты с заданной точностью для заданного  $x$  по формуле

3. Написать программу для печати всех слов из заданного текста (символьной строки) в обратном порядке (от последнего к первому)

БИЛЕТ 4

1. Массивы

2. Написать программу для обхода дерева с рекурсией

3. В тексте (символьной строке) находятся целые числа без знака в символьном виде.

Написать программу, которая найдет и напечатает их сумму.

БИЛЕТ 5

1. Символьные строки

2. Написать программу для нахождения минимума и максимума последовательности целых чисел без массивов

3. Написать программу, которая напечатает линейный список в обратном порядке

БИЛЕТ 6

1. Основные типы операторов

2. Дана символьная строка. Написать программу для печати каждого слова буквами наоборот

3. Написать программу для нахождения пересечения упорядоченных массивов целых чисел

БИЛЕТ 7

1. Сортировки и поиск, слияние

2. Написать программу для вычисления факториала с рекурсией и без

3. Написать программу, которая в квадратной матрице вычислит сумму элементов по всем диагоналям одного направления

БИЛЕТ 8

1. Область видимости и время жизни объектов

2. Написать программу для вычисления чисел Фибоначчи с рекурсией и без

3. Написать программу, которая в целочисленном массиве найдет наибольшую сумму подряд идущих элементов

БИЛЕТ 9

1. Классы

2. Написать программу для упорядочения массива целых чисел путем сравнением соседних ("пузырьком")

3. Написать программу, которая напечатает из текста (символьной строки) все слова по алфавиту

БИЛЕТ 10

1. Структуры

2. Дана символьная строка. Написать программу для печати каждого слова с новой строки и его длины

3. Написать программу, которая в квадратной матрице вычислит сумму элементов, которые находятся или под

главной или под побочной диагональю

БИЛЕТ 11

1. Линейные списки

2. Написать программу для упорядочения массива целых чисел путем нахождения минимума

3. Написать программу, которая в тексте найдет и напечатает все слова, которые начинаются с заглавной буквы и

не содержат ничего, кроме латинских букв

БИЛЕТ 12

1. Рекурсия

2. Написать программу для вычисления значения многочлена для заданного значения аргумента  $x$

3. Написать программу для нахождения разности двух упорядоченных массивов целых чисел

БИЛЕТ 13

1. Стек, очередь, ДЭК

2. Дана символьная строка. Написать программу для печати самого длинного слова

3. Написать программу, которая в прямоугольной матрице вычислит сумму элементов, которые находятся на одной

вертикали, горизонтали или диагонали с заданным элементом

БИЛЕТ 14

1. Деревья

2. Написать программу для умножения двух прямоугольных матриц

3. Написать программу для нахождения длины самой длинной возрастающей цепочки элементов в

последовательности чисел (без массивов)

БИЛЕТ 15

1. Файловый ввод-вывод

2. Написать программу для проверки упорядоченности последовательности целых чисел (без массива)

3. Написать программу, которая в квадратной матрице вычислит сумму элементов, которые находятся

одновременно и под главной и под побочной диагоналями

БИЛЕТ 16

1. Прямой доступ к файлам

2. Написать программу для проверки того, что элементы массива образуют арифметическую прогрессию

### 3. Написать программу для поворота квадратной матрицы на 180 градусов

#### БИЛЕТ 17

1. Многомерные массивы
2. Написать программу, которая в заданном массиве сдвинет все положительные элементы к началу, не меняя их взаимного расположения
3. Написать программу, которая в двух символьных строках найдет и напечатает самую длинную одинаковую цепочку подряд идущих символов

#### БИЛЕТ 18

1. Данные, их типы и описание
2. Написать программу для умножения двух заданных многочленов
3. Написать функцию для перевода целого числа в двоичную систему

### 7.1. Основная литература:

1. Информатика: Курс лекций. Учебное пособие / Е.Л. Федотова, А.А. Федотов. - М.: ИД ФОРУМ: ИНФРА-М, 2011. - 480 с.  
<http://znanium.com/catalog.php?bookinfo=204273>
2. Информатика. Базовый курс : учебное пособие для студентов высших технических учебных заведений / под ред. С. В. Симоновича. ? 2-е изд. ? Санкт-Петербург [и др.] : Питер, 2008. ? 639 с. :
3. Андрианова, А.А. Практикум по курсу "Алгоритмизация и программирование". Часть 1 / А.А. Андрианова, Т.М. Мухтарова. - Казань: Казанский государственный университет, 2008. - 96 с.
4. Андрианова А.А., Мухтарова Т.М. Практикум по курсу "Алгоритмизация и программирование" - часть 1. - Казанский государственный университет, 2008.  
[http://libweb.ksu.ru/ebooks/09\\_63.pdf](http://libweb.ksu.ru/ebooks/09_63.pdf)
5. Практикум работы на ЭВМ: учебное пособие / [В. С. Кугураков и др.]; Казан. гос. ун - т, Фак. вычисл. математики и кибернетики. - Казань: КГУ, 2008. - 111 с.
6. Могилев, А. В. Методы программирования. Компьютерные вычисления / А. В. Могилев, Л. В. Листрова. ? СПб.: БХВ-Петербург, 2008. ? 320 с. URL:  
<http://znanium.com/bookread.php?book=350418>

### 7.2. Дополнительная литература:

1. Андрианова А.А., Мухтарова Т.М. Объектно-ориентированный анализ и программирование. Конспект лекций. - Казан. федер. ун-т, Казань, 2013. - 137 с.  
[http://libweb.kpfu.ru/ebooks/09-IVMIT/09\\_104\\_kl-000497.pdf](http://libweb.kpfu.ru/ebooks/09-IVMIT/09_104_kl-000497.pdf)
2. Каймин В.А. Информатика. - М.: ИНФРА-М, 2010. - 285 с.  
<http://znanium.com/bookread.php?book=224852>
3. Гвоздева В.А. Информатика, автоматизированные информационные технологии и системы. - М.: ИНФРА-М, 2011. - 544 с.  
<http://znanium.com/bookread.php?book=207105>

### 7.3. Интернет-ресурсы:

- Интернет-портал образовательных ресурсов КФУ - <http://www.kfu-elearning.ru/>  
Интернет-портал образовательных ресурсов по ИТ - <http://www.intuit.ru>  
Интернет-портал с ресурсами по программным продуктам компании Microsoft - <http://www.msdn.ru>



Интернет-портал со статьями по алгоритмике и программированию - <http://algolist.manual.ru/>  
Электронная библиотека по техническим наукам - <http://techlibrary.ru>

## **8. Материально-техническое обеспечение дисциплины(модуля)**

Освоение дисциплины "Технологии и методы программирования" предполагает использование следующего материально-технического обеспечения:

Мультимедийная аудитория, вместимостью более 60 человек. Мультимедийная аудитория состоит из интегрированных инженерных систем с единой системой управления, оснащенная современными средствами воспроизведения и визуализации любой видео и аудио информации, получения и передачи электронных документов. Типовая комплектация мультимедийной аудитории состоит из: мультимедийного проектора, автоматизированного проекционного экрана, акустической системы, а также интерактивной трибуны преподавателя, включающей тач-скрин монитор с диагональю не менее 22 дюймов, персональный компьютер (с техническими характеристиками не ниже Intel Core i3-2100, DDR3 4096Mb, 500Gb), конференц-микрофон, беспроводной микрофон, блок управления оборудованием, интерфейсы подключения: USB, audio, HDMI. Интерактивная трибуна преподавателя является ключевым элементом управления, объединяющим все устройства в единую систему, и служит полноценным рабочим местом преподавателя. Преподаватель имеет возможность легко управлять всей системой, не отходя от трибуны, что позволяет проводить лекции, практические занятия, презентации, вебинары, конференции и другие виды аудиторной нагрузки обучающихся в удобной и доступной для них форме с применением современных интерактивных средств обучения, в том числе с использованием в процессе обучения всех корпоративных ресурсов. Мультимедийная аудитория также оснащена широкополосным доступом в сеть интернет. Компьютерное оборудование имеет соответствующее лицензионное программное обеспечение.

Компьютерный класс, представляющий собой рабочее место преподавателя и не менее 15 рабочих мест студентов, включающих компьютерный стол, стул, персональный компьютер, лицензионное программное обеспечение. Каждый компьютер имеет широкополосный доступ в сеть Интернет. Все компьютеры подключены к корпоративной компьютерной сети КФУ и находятся в едином домене.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе "ZNANIUM.COM", доступ к которой предоставлен студентам. ЭБС "ZNANIUM.COM" содержит произведения крупнейших российских учёных, руководителей государственных органов, преподавателей ведущих вузов страны, высококвалифицированных специалистов в различных сферах бизнеса. Фонд библиотеки сформирован с учетом всех изменений образовательных стандартов и включает учебники, учебные пособия, УМК, монографии, авторефераты, диссертации, энциклопедии, словари и справочники, законодательно-нормативные документы, специальные периодические издания и издания, выпускаемые издательствами вузов. В настоящее время ЭБС ZNANIUM.COM соответствует всем требованиям федеральных государственных образовательных стандартов высшего профессионального образования (ФГОС ВПО) нового поколения.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе Издательства "Лань", доступ к которой предоставлен студентам. ЭБС Издательства "Лань" включает в себя электронные версии книг издательства "Лань" и других ведущих издательств учебной литературы, а также электронные версии периодических изданий по естественным, техническим и гуманитарным наукам. ЭБС Издательства "Лань" обеспечивает доступ к научной, учебной литературе и научным периодическим изданиям по максимальному количеству профильных направлений с соблюдением всех авторских и смежных прав.

Компьютерные классы лаборатории малой вычислительной техники Института ВМ и ИТ, оснащенные мультимедийным оборудованием.

Программа составлена в соответствии с требованиями ФГОС ВПО и учебным планом по направлению 10.03.01 "Информационная безопасность" и профилю подготовки Безопасность автоматизированных систем .



Автор(ы):

Васильев А.В. \_\_\_\_\_

Андрианова А.А. \_\_\_\_\_

"\_\_" \_\_\_\_\_ 201\_\_ г.

Рецензент(ы):

Тагиров Р.Р. \_\_\_\_\_

"\_\_" \_\_\_\_\_ 201\_\_ г.