

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Национальный исследовательский ядерный университет «МИФИ»

**Саровский физико-технический институт -**

филиал федерального государственного автономного образовательного учреждения высшего  
образования «Национальный исследовательский ядерный университет «МИФИ»  
(СарФТИ НИЯУ МИФИ)

**ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И ЭЛЕКТРОНИКИ**  
**Кафедра «Вычислительной и информационной техники»**

**УТВЕРЖДАЮ**

Декан ФИТЭ, к.ф-м.н., доцент

\_\_\_\_\_ **В.С. Холушкин**

«\_\_» \_\_\_\_\_ **2023 г.**

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ С ОТКРЫТЫМ КОДОМ**

наименование дисциплины

Направление подготовки (специальность)	09.03.01 Информатика и вычислительная техника
Наименование образовательной программы	Программное и аппаратное обеспечение высокопроизводительных вычислительных систем и сетей
Квалификация (степень) выпускника	бакалавр
Форма обучения	очная

Программа одобрена на заседании кафедры

Зав. кафедрой ВИТ

Протокол № \_\_\_\_\_ от \_\_\_\_\_

\_\_\_\_\_ **В.С. Холушкин**

«\_\_» \_\_\_\_\_ **2023г.**

г. Саров, 2023г.

Программа переутверждена на 202\_\_\_\_/202\_\_\_\_ учебный год с изменениями в соответствии с семестровыми учебными планами академических групп ФТФ, ФИТЭ на 202\_\_\_\_/202\_\_\_\_ учебный год.

Заведующий кафедрой ВИТ

В.С. Холушкин

Программа переутверждена на 202\_\_\_\_/202\_\_\_\_ учебный год с изменениями в соответствии с семестровыми учебными планами академических групп ФТФ, ФИТЭ на 202\_\_\_\_/202\_\_\_\_ учебный год.

Заведующий кафедрой ВИТ

В.С. Холушкин

Программа переутверждена на 202\_\_\_\_/202\_\_\_\_ учебный год с изменениями в соответствии с семестровыми учебными планами академических групп ФТФ, ФИТЭ на 202\_\_\_\_/202\_\_\_\_ учебный год.

Заведующий кафедрой ВИТ

В.С. Холушкин

Программа переутверждена на 202\_\_\_\_/202\_\_\_\_ учебный год с изменениями в соответствии с Семестровыми учебными планами академических групп ФТФ, ФИТЭ на 202\_\_\_\_/202\_\_\_\_ учебный год.

Заведующий кафедрой ВИТ

В.С. Холушкин

Семестр	В форме прак- тической подго- товки	Трудоемкость, кред.	Общий объем курса, час.	Лекции, час.	Практич. заня- тия, час.	Лаборат. работы, час.	СРС, час.	КР/ КП	Форма(ы) кон- троля, экз./зач./ЗсО/
2	32	4	144	16	-	48	44	-	Э
<b>ИТОГО</b>	<b>32</b>	<b>4</b>	<b>144</b>	<b>16</b>	<b>-</b>	<b>48</b>	<b>44</b>	<b>-</b>	<b>36</b>

## **АННОТАЦИЯ**

Курс посвящен изучению теоретических и практические основ алгоритмизации и программирования кроссплатформенных приложений. Изучаются способы и методы разработки эффективных алгоритмов для решения различных прикладных задач. Главная цель преподавания дисциплины – подготовка специалиста, владеющего фундаментальными знаниями и практическими навыками в области основ алгоритмизации и программирования для решения прикладных задач в различных предметных областях.

### **1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ**

В дисциплине «Программное обеспечение с открытым кодом» рассматриваются теоретические и практические аспекты основ программирования кроссплатформенных приложений. Изучаются способы и методы разработки эффективных алгоритмов для решения различных прикладных задач. Рассматриваются общие принципы создания программ на языке C++ с использованием библиотеки Qt. Главная цель преподавания дисциплины – подготовка специалиста, владеющего фундаментальными знаниями и практическими навыками в области программирования.

Задачи дисциплины:

- Дать представление о программном обеспечении с открытым кодом
- Сформировать навыки создания кроссплатформенных приложений с графическим интерфейсом.

### **2. МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО**

Дисциплина «Программное обеспечение с открытым кодом» является вариативной дисциплиной общепрофессиональной части ООП и базируется на таких дисциплинах как, «Информатика», «Информационные технологии», «Основы алгоритмизации и программирования», «Технологии программирования».

Освоение дисциплины «Программное обеспечение с открытым кодом» необходимо для последующего применения полученных знаний в профессиональной деятельности, а также для успешного выполнения производственной практики и научно-исследовательской работы бакалавра.

### 3.ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

#### Общепрофессиональные компетенции (ОПК)

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
<b>ОПК-8</b> Способен разрабатывать алгоритмы и программы, пригодные для практического применения	<b>З-ОПК-8</b> Знать: алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения <b>У-ОПК-8</b> Уметь: составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули <b>В-ОПК-8</b> Владеть: языком программирования; навыками отладки и тестирования работоспособности программы
<b>ОПК-2</b> Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	<b>З-ОПК-2</b> Знать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, используемых при решении задач профессиональной деятельности <b>У-ОПК-2</b> Уметь выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности <b>В-ОПК-2</b> Владеть навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности

#### 4.СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

№ п/п	Наименование раздела /темы дисциплины	№ не- дели	Виды учебной работы					
			Лекции	Практ. занятия/ семинары	Лаб. работы	СРС	Текущий контроль (форма)*	Максимальный балл (см. п. 5.3)
			16	-	48	44		
Семестр 2								
Раздел 1.								
1.1.	Тема 1. . Введение. Ис- тория появления ПО с открытым кодом. Исто- рия появления библиоте- ки для создания крос- сплатформенных прило- жений.	1	2	-	4	5	УО, Защи- та ЛР	4

№ п/п	Наименование раздела /темы дисциплины	№ не- дели	Виды учебной работы					
			Лекции	Практ. занятия/ семинары	Лаб. работы	СРС	Текущий контроль (форма)*	Максимальный балл (см. п. 5.3)
			16	-	48	44		
1.2	Тема 2. Основные прин- ципы создания приложе- ний. Обработка событий. Создание диалоговых приложений. Решение простых задач.	2	2	-	6	5	Защита ЛР	4
Раздел 2.								
2.1	Тема 1. Отладка и тести- рование программ.	3,4	2	-	6	5	Защита ЛР	4
2.2	Тема 2. Решение задач с использованием вспомо- гательных функций.	5,6	2	-	6	5	Защита ЛР	4
2.3	Тема 3 Регулярные вы- ражения	7,8	2	-	6	6	Защита ЛР	4
2.4	Тема 4.Решение задач по обработке строк и ис- пользованием класса QString	9,10	2	-	6	6	Защита ЛР	4
Рубежный контроль		11	СР					7
Раздел 3.								
3.1	Тема 1. Работа с файлами	12,13	2	-	6	6	Защита ЛР	4
3.2	Тема 2. Основные сред- ства для построения ри- сунков. Обработка нажа- тия клавиш на клавиатуре и событий мыши.	14,15	2	-	6	6	Защита ЛР	4
Рубежный контроль		16	СР					10
Промежуточная аттестация			Экзамен				-	50
Посещаемость								5
Итого:			16	-	48	44	-	100

\*Сокращение наименований форм текущего, рубежного и промежуточного контроля:

УО – устный опрос

СР – самостоятельная работа(решение задачи на заданную тему)

РГР – расчетно – графическая работа

#### 4.2. Содержание дисциплины, структурированное по разделам (темам)

##### Лекционный курс

№	Наименование раздела /темы дисциплины	Содержание
<b>2 семестр</b>		
<b>Раздел 1</b>		
1.1	Тема 1. Введение. История появления ПО с от-	История появления ПО с открытым кодом. История появления библиотеки для создания кроссплатформенных

	крытым кодом. История появления библиотеки для создания кросс-платформенных приложений.	приложений. Обзор библиотек для создания приложений на языке C++.
1.2	Тема 2. Основные принципы создания приложений. Обработка событий. Создание диалоговых приложений. Решение простых задач.	Основы работы в среде QtCreator. Обработка событий. Создание диалоговых приложений. Примеры решения простых задач.
<b>Раздел 2</b>		
2.1	Тема 1. Отладка и тестирование программ	Средства отладки: метод qDebug(), пошаговое выполнение программы, просмотр значений переменных. Пример тестирования программы.
2.2	Тема 2. Решение задач с использованием вспомогательных функций.	Примеры решения задач с помощью приложений, использующих вспомогательные функции
2.3	Тема 3. Регулярные выражения	Назначение регулярных выражений. Правила записи. Примеры применения.
2.4	Тема 4. Решение задач по обработке строк и использованием класса QString	Основные методы класса QString. Примеры их использования для обработки строк.
<b>Раздел 3.</b>		
3.1	Тема 1. Работа с файлами	Класс QFile. Потоки. Специальные окна для работы с файлами. Пример обработки текстового файла..
3.2	Тема 2. . Основные средства для построения рисунков. Обработка нажатия клавиш на клавиатуре и событий мыши.	Методы класса QPainter. Перо, кисть, графические примитивы. Пример программы. Основы обработки событий клавиатуры и мыши. Перегрузка виртуальных функций. Примеры приложений.

## Практические занятия

№ п/п	Раздел дисциплины – второй семестр
1	Введение. История появления ПО с открытым кодом. История появления библиотеки для создания кроссплатформенных приложений.
2	Основные принципы создания приложений. Обработка событий. Создание диалоговых приложений. Решение простых задач.
3	Отладка и тестирование программ.
4	Решение задач с использованием вспомогательных функций.
5	Регулярные выражения.
6	Решение задач по обработке строк и использованием класса QString
7	Работа с файлами
8	Основные средства для построения рисунков.
9	Обработка нажатия клавиш на клавиатуре и событий мыши.

## Лабораторные занятия

Лабораторный практикум предполагает выполнение лабораторных работ по основным разделам дисциплины. Темы лабораторных работ приведены в следующей таблице. Инструментальная среда для выполнения лабораторных работ: QtCreator.

№	Примерные темы лабораторных занятий
1	Создание простых диалоговых приложений
2	Отладка и тестирование программ
3	Решение задач с использованием вспомогательных функций.
4	Приложения для обработки строк
5	Работа с файлами
6	Приложения, использующие графические элементы
7	Рисование с помощью мыши
8	Области отсечения
9	Использование класса QVector. Области прокрутки. Построение круговой диаграммы



#### **4.3 Перечень учебно-методического обеспечения для самостоятельной работы студентов**

При изучении дисциплины используются следующие виды самостоятельной работы:

- самостоятельный поиск литературы по разделам и темам курса;
- изучение материала по дополнительным разделам дисциплины;
- изучение литературы и подготовка к выполнению лабораторных работ, курсовых работ;
- подготовка к тестированию, контрольным работам, написанию рефератов;
- подготовка к зачету, экзаменам.

Форма контроля: отчет по лабораторным работам и их защита, защита курсовых работ.

#### **Учебно-методические пособия:**

1. Шлее М. Qt4. Профессиональное программирование на C++. СПб.: БХВ-Петербург, 2007.
2. Травова Н.Н. Лабораторный практикум по программированию в среде Qt(размещен на Server-129 в папке public/Travova/Qt).
3. Бланшет Ж., Саммерфилд М. Qt4. Программирование GUI на C++. 2-е изд. М.: КУДИЦ-Пресс, 2007.
4. Земсков Ю. Qt4 на примерах. СПб.: БХВ-Петербург, 2008.

### **5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

Фонд оценочных средств по дисциплине обеспечивает проверку освоения планируемых результатов обучения (компетенций и их индикаторов) посредством мероприятий текущего, рубежного и промежуточного контроля по дисциплине.

#### **5.1. Паспорт фонда оценочных средств по дисциплине**

Связь между формируемыми компетенциями и формами контроля их освоения представлена в следующей таблице:

Раздел	Темы занятий	Компетенция	Индикаторы освоения	Текущий контроль, неделя
<b>Семестр 2</b>				
Раздел 1	Тема 1. . Введение. История появления ПО с открытым кодом. История появления библиотеки для создания кроссплатформенных приложений.	ОПК-2,ОПК-8	3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	УО 1
	Тема 2. Основные принципы создания приложений. Обработка событий. Создание диалоговых приложений. Решение простых задач		3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	Защита ЛР3
Раздел 2	Тема 1. Отладка и тестирование программ.	ОПК-2,ОПК-8	3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	Защита ЛР5
	Тема 2. Решение задач с использованием вспомогательных функций.		3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	Защита ЛР7
	Тема 3 Регулярные выражения		3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	Защита ЛР9
	Тема 4.Решение задач по обработке строк и использованием класса QString		3-ОПК-8; У-ОПК-8; В-ОПК-8 3-ОПК-9; У-ОПК-9; В-ОПК-9	Защита ЛР10
<b>Рубежный контроль</b>		ОПК-2,ОПК-8	3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	СР 11
Раздел 3	Тема 1. Работа с файлами	ОПК-2,ОПК-8	3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	Защита ЛР13
	Тема 2. Основные средства для построения рисунков. Обработка нажатия клавиш на клавиатуре и событий мыши.		3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	Защита ЛР 15
<b>Рубежный контроль</b>		ОПК-2,ОПК-8	3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	
<b>Промежуточная аттестация</b>		ОПК-2,ОПК-8	3-ОПК-2; У-ОПК-2; В-ОПК-2 3-ОПК-8; У-ОПК-8; В-ОПК-8	<b>Экзамен</b>

**5.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения образовательной программы**

**5.2.1. Оценочные средства для текущего контроля**

**5.2.1.1. Примерные вопросы для устного опроса (УО)**

1. Создание простых диалоговых приложений
2. Отладка и тестирование программ
3. Решение задач с использованием вспомогательных функций.
4. Приложения для обработки строк
5. Работа с файлами

6. Приложения, использующие графические элементы
7. Рисование с помощью мыши
8. Использование класса QVector. Области прокрутки. Построение круговой диаграммы

#### 5.2.1.2. Примерные темы и вопросы для самостоятельной работы (СР)

1. Использование языка программирования C++ для решения прикладных задач из различных предметных областей (математика, физика, экономика и др.)
2. Использование библиотек при программировании на C, C++ для решения задач со сложными типами данных (очереди, стеки, множества, списки и др.)
3. Особенности программирования консольных приложений. Программные средства создания графических интерфейсов.

#### 5.2.2. Оценочные средства для рубежного контроля

##### 5.2.2.1. Примерные задания для решения задач по заданной теме

1. Создайте приложение, которое позволяет рисовать **отрезки прямых** линий с помощью мыши. Создаваемая линия должна быть видима в процессе построения. Цвет линий должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить интенсивности красной, зеленой и синей составляющих цвета. Предоставьте возможность сохранять изображение в файле и считывать его из файла.
2. Создайте приложение, которое позволяет рисовать **закрашенные прямоугольники** с помощью мыши. Создаваемый прямоугольник должен быть видим в процессе построения. Цвет прямоугольников должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить интенсивности красной, зеленой и синей составляющих цвета. Предоставьте возможность сохранять изображение в файле и считывать его из файла.
3. Создайте приложение, которое позволяет рисовать **закрашенные треугольники**, задаваемые тремя щелчками мыши в вершинах. Цвет треугольников должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить интенсивности красной, зеленой и синей составляющих цвета. Предоставьте возможность сохранять изображение в файле и считывать его из файла.

4. Создайте приложение, которое позволяет рисовать **закрашенные круги** с помощью мыши. Создаваемая окружность должна быть видима в процессе построения. Цвет окружностей должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить интенсивности красной, зеленой и синей составляющих цвета. Предоставьте возможность сохранять изображение в файле и считывать его из файла.
5. Создайте приложение, которое рисует закрашенные **пятиконечные звезды** (вписанную в воображаемую окружность; центр окружности и одна точка на окружности задаются щелчками мыши). Цвет каждой звездочки должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить интенсивности красной, зеленой и синей составляющих цвета. Предоставьте возможность сохранять изображение в файле и считывать его из файла.
6. Создайте приложение, которое рисует **закрашенные квадраты** с помощью мыши так, что мышь проходит по диагонали квадрата от одной вершины к другой. Квадрат должен быть виден в процессе построения. Цвет линий должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить интенсивности красной, зеленой и синей составляющих цвета. Предоставьте возможность сохранять изображение в файле и считывать его из файла.
7. Создайте приложение, которое рисует **дуги** с помощью мыши так, что мышь определяет начало и конец дуги. Дуга должен быть видима в процессе построения. Цвет линий должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить интенсивности красной, зеленой и синей составляющих цвета. Предоставьте возможность сохранять изображение в файле и считывать его из файла.
8. Создайте приложение, которое рисует **правильные треугольники** с помощью мыши так, что указатель мыши находится в одной из вершин треугольника. Треугольник должен быть видим в процессе построения и может поворачиваться. Цвет линий должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить иненсивности красной,

зеленой и синей составляющих цвета. При отпускании мыши треугольник **закрашивается**. Предоставьте возможность сохранять изображение в файле и считывать его из файла.

9. Создайте приложение, которое рисует **правильные 8-конечные звездочки** с помощью мыши так, что указатель мыши находится в одной из вершин звездочки. Звездочка должна быть видима в процессе построения и может поворачиваться. При отпускании мыши звездочка **закрашивается**. Цвет линий должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить интенсивности красной, зеленой и синей составляющих цвета. Предоставьте возможность сохранять изображение в файле и считывать его из файла.
10. Создайте однодокументное приложение, которое рисует **правильные 5-конечные звездочки** с помощью мыши так, что указатель мыши находится в одной из вершин звездочки. Звездочка должна быть видима в процессе построения и может поворачиваться. Цвет линий должен определяться с помощью **немодального** диалогового окна с наборными счетчиками (Spin), которые позволяют установить интенсивности красной, зеленой и синей составляющих цвета. Предоставьте возможность сохранять изображение в файле и считывать его из файла.

### 5.2.3. Оценочные средства для промежуточной аттестации

#### 5.2.3.2. Примерные вопросы к экзамену

1. История появления ПО с открытым кодом.
2. История появления библиотеки для создания кроссплатформенных приложений.
3. Обзор библиотек для создания приложений на языке C++.
4. Основные принципы создания приложений
5. Основы работы в среде QtCreator.
6. Обработка событий.
7. Создание диалоговых приложений. Примеры решения простых задач.
8. Отладка и тестирование программ
9. Средства отладки: метод `qDebug()`, пошаговое выполнение программы, просмотр значений переменных.
10. Пример тестирования программы.
11. Решение задач с использованием вспомогательных функций

12. Примеры решения задач с помощью приложений, использующих вспомогательные функции.
13. Регулярные выражения.
14. Назначение регулярных выражений. Правила записи. Примеры применения.
15. Решение задач по обработке строк и использованием класса QString
16. Основные методы класса QString.
17. Примеры использования методов класса QString для обработки строк.
18. Работа с файлами
19. Класс QFile.
20. Потоки.
21. Специальные окна для работы с файлами.
22. Пример обработки текстового файла.
23. Основные средства для построения рисунков.
24. Методы класса QPainter.
25. Перо, кисть, графические примитивы. Пример программы.
26. Обработка нажатия клавиш на клавиатуре и событий мыши
27. Основы обработки событий клавиатуры и мыши.
28. Перегрузка виртуальных функций. Примеры приложений.

### 5.3. Шкалы оценки образовательных достижений

Рейтинговая оценка знаний является интегральным показателем качества теоретических и практических знаний и навыков студентов по дисциплине и складывается из оценок, полученных в ходе текущего контроля и промежуточной аттестации.

Результаты текущего контроля и промежуточной аттестации подводятся по шкале балльно-рейтинговой системы.

Шкала каждого контрольного мероприятия лежит в пределах от 0 до установленного максимального балла включительно. Итоговая аттестация по дисциплине оценивается по 100-балльной шкале и представляет собой сумму баллов, заработанных студентом при выполнении заданий в рамках текущего и промежуточного контроля.

Итоговая оценка выставляется в соответствии со следующей шкалой:

Сумма баллов	Оценка по 4-ех балльной шкале	Оценка ECTS	Требования к уровню освоению учебной дисциплины
90-100	5 – <i>«отлично»</i>	A	Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно,

			четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, использует в ответе материал монографической литературы.
85-89	4 – «хорошо»	B	Оценка «хорошо» выставляется студенту, если он твёрдо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос.
75-84		C	
70-74		D	
65-69	3 – «удовлетворительно»	E	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала.
60-64			
Ниже 60	2 – «неудовлетворительно»	F	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

## 6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 6.1. Рекомендуемая литература

#### ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ:

1. Шлее М. Qt4. Профессиональное программирование на C++. СПб.: БХВ-Петербург, 2007.
2. Травова Н.Н. Лабораторный практикум по программированию в среде Qt(размещен на Server-129 в папке public/Travova/Qt).
3. Бланшет Ж., Саммерфилд М. Qt4. Программирование GUI на C++. 2-е изд. М.: КУДИЦ-Пресс, 2007.
4. Земсков Ю. Qt4 на примерах. СПб.: БХВ-Петербург, 2008.
5. Шилдт Г. Полный справочник по C++. 4-е изд. М.: Вильямс, 2006
6. Стефенс Д. Р. C++. Сборник рецептов. М.: КУДИЦ-Пресс, 2007.

7. Саммерфилд М. Qt4. Профессиональное программирование. Разработка кросс-платформенных приложений на C++. СПб.: Символ-Плюс 2011.
8. Дж.Макконнелл Анализ алгоритмов. Вводный курс. М.: ТЕХНОСФЕРА, 2002.

## **6.2. Средства обеспечения освоения дисциплины**

Учебно-методический материал в библиотеке института, ресурсы Интернета, ресурсы электронной библиотеки

## **7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

Класс ПЭВМ не ниже Intel Pentium 4, 512M RAM, 40G HDD с установленным программным обеспечением: MS WindowsXP, MS Office Pro, Microsoft Visual Studio 613 и др.из расчета одна ПЭВМ на одного человека. Класс ПЭВМ не ниже IntelPentium 4, 512MRAM, 40GHDD с установленным программным обеспечением: OCLinuxс графической оболочкой GNOM, QtCreator

## **8.ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ**

В соответствии с требованиями ОС ВО по направлению подготовки 09.03.01 «Информатика и вычислительная техника» реализация компетентностного подхода предусматривает широкое использование в учебном процессе активных и интерактивных форм проведения занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков студентов. В рамках учебного курса студенты работают с лекциями, рекомендованной литературой, выполняют лабораторные работы, готовятся к экзамену и зачету. В процессе подготовки студенты используют программные продукты, инструментальные среды, информационно-справочные системы, информационные источники, размещенные в сети Интернет (официальные сайты, веб-порталы, тематические форумы и телекоммуникации), электронные учебники и учебно-методические пособия.

## **9. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ СТУДЕНТАМ ПО ОРГАНИЗАЦИИ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ**

### **Предлагается**

- Самостоятельно прорабатывать лекционный материал для более полного усвоения материала;
- В учебном процессе при выполнении лабораторного практикума эффективно использовать методические пособия и методический материал по темам лабораторных работ;
- Активно использовать Интернет-ресурсы для получения актуального материала по изучаемой дисциплине;



- Активно использовать Интернет-ресурсы для обновления инструментальной базы (систем программирования, инструментальных сред и т.д.) при выполнении лабораторных работ.

Программа составлена в соответствии с требованиями ОС ВО НИЯУ МИФИ к обязательному минимуму содержания основной образовательной программы по направлению подготовки 09.03.01 Информатика и вычислительная техника.

Автор(ы) \_\_\_\_\_ Н.Н.Травова

Рецензенты \_\_\_\_\_ В.С.Холушкин

Согласовано:

Зав. кафедрой ВИТ \_\_\_\_\_ В.С.Холушкин

Руководитель ОП \_\_\_\_\_ В.С.Холушкин