

Практикум по дисциплине

**КОМПЬЮТЕРНОЕ ОБЕСПЕЧЕНИЕ
РАСЧЕТНО-ПРОЕКТНОЙ
И ЭКСПЕРИМЕНТАЛЬНО-ИССЛЕДОВАТЕЛЬСКОЙ
ДЕЯТЕЛЬНОСТИ**

Последовательность рассматриваемых тем

Тема 1 Компьютерное моделирование

Тема 2 Компьютерная обработка экспериментальных данных

Тема 3 Работа с графическими данными

Тема 4 Средства разработки приложений

Краткое изложение содержания каждой темы

Тема 1 Компьютерное моделирование

Компьютерные модели стали обычным инструментом математического моделирования и применяются в физике, астрофизике, механике, химии, биологии, экономике, социологии, метеорологии, других науках и прикладных задачах в различных областях радиоэлектроники, машиностроения, автомобилестроения и проч. Компьютерные модели используются для получения новых знаний об объекте или для приближенной оценки поведения систем, слишком сложных для аналитического исследования.

Компьютерное моделирование является одним из эффективных методов изучения сложных систем. Компьютерные модели проще и удобнее исследовать в силу их возможности проводить т. н. вычислительные эксперименты, в тех случаях когда реальные эксперименты затруднены из-за финансовых или физических препятствий или могут дать непредсказуемый результат. Логичность и формализованность компьютерных моделей позволяет определить основные факторы, определяющие свойства изучаемого объекта-оригинала (или целого класса объектов), в частности, исследовать отклик моделируемой физической системы на изменения её параметров и начальных условий.

Построение компьютерной модели базируется на абстрагировании от конкретной природы явлений или изучаемого объекта-оригинала и состоит из двух этапов — сначала создание качественной, а затем и количественной модели. Чем больше значимых свойств будет выявлено и перенесено на компьютерную модель — тем более приближенной она окажется к реальной модели, тем большими возможностями сможет обладать система, использующая данную модель. Компьютерное же моделирование заключается в проведении серии вычислительных экспериментов на компьютере, целью которых является анализ, интерпретация и

сопоставление результатов моделирования с реальным поведением изучаемого объекта и, при необходимости, последующее уточнение модели и так далее

Различают аналитическое и имитационное моделирование. При аналитическом моделировании изучаются математические (абстрактные) модели реального объекта в виде алгебраических, дифференциальных и других уравнений, а также предусматривающих осуществление однозначной вычислительной процедуры, приводящей к их точному решению. При имитационном моделировании исследуются математические модели в виде алгоритма(ов), воспроизводящего функционирование исследуемой системы путём последовательного выполнения большого количества элементарных операций.

Компьютерное моделирование дает возможность:

расширить круг исследовательских объектов - становится возможным изучать не повторяющиеся явления, явления прошлого и будущего, объекты, которые не воспроизводятся в реальных условиях;

визуализировать объекты любой природы, в том числе и абстрактные;

исследовать явления и процессы в динамике их развертывания;

управлять временем (ускорять, замедлять и т.д.);

совершать многократные испытания модели, каждый раз возвращая её в первичное состояние;

получать разные характеристики объекта в числовом или графическом виде;

находить оптимальную конструкцию объекта, не изготавливая его пробных экземпляров;

проводить эксперименты без риска негативных последствий для здоровья человека или окружающей среды.

Название этапа	Исполнение действий
1. Постановка задачи и её анализ	1.1. Выяснить, с какой целью создается модель. 1.2. Уточнить, какие исходные результаты и в каком виде следует их получить. 1.3. Определить, какие исходные данные нужны для создания модели.
2. Построение информационной модели	2.1. Определить параметры модели и выявить взаимосвязь между ними. 2.2. Оценить, какие из параметров влиятельные для данной задачи, а какими можно пренебрегать. 2.3. Математически описать зависимость между параметрами модели.
3. Разработка метода и алгоритма реализации компьютерной модели	3.1. Выбрать или разработать метод получения исходных результатов. 3.2. Составить алгоритм получения результатов по избранным методам. 3.3. Проверить правильность алгоритма.
4. Разработка компьютерной модели	4.1. Выбрать средства программной реализации алгоритма на компьютере. 4.2. Разработать компьютерную модель. 4.3. Проверить правильность созданной компьютерной модели.
5. Проведение эксперимента	5.1. Разработать план исследования. 5.2. Провести эксперимент на базе созданной компьютерной модели. 5.3. Проанализировать полученные результаты. 5.4. Сделать выводы насчет свойств прототипа модели.

В процессы проведения эксперимента может выясниться, что нужно:

скорректировать план исследования;

выбрать другой метод решения задачи;

усовершенствовать алгоритм получения результатов;

уточнить информационную модель;

внести изменения в постановку задачи.

Задание на практическую работу

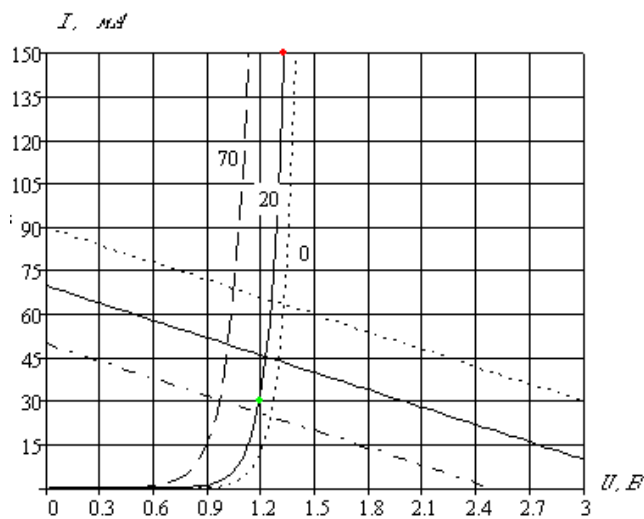
Самостоятельно провести компьютерное моделирование характеристик лазерного диода на основании исходных данных, приведенных в таблице

Параметр	Значение	
Ток насыщения, мкА	0,035	
Коэф. в показатели степени	3,4	
Темп. удв. темного тока, °С	7,8	
Рабочий ток, мА		
Рабочее напряжение, В	1,35	
Мах. прямой ток, мА	150	
Мах. прямое напряжение, В	1,4	
Пороговая мощность P _{th} , мВт	0,05	
Рабочая мощность, P _{op} , мВт	2	
Излучательная характеристика	I _{th} , мА	I _{op} , мА
-100	10	25
-80	10	25
-60	10	25
-20	10	25
0	11	26
25	13	30
40	17	34
60	23	42
70	35	60
100	50	100
Рабочая T _{max} /T _{min} , °С	0 / +70	
Характер. деград. час	10 x 10 ⁵	
Время нарастания, пс	40	
Время спада, пс	90	
Постоянная времени, пс	60	
Частота среза, ГГц	4	
Полуширина спектра, нм	0,02	
Материал	InGaASP (DFB)	
Номер образца	2	
Марка прототипа	NDL 7705 P	

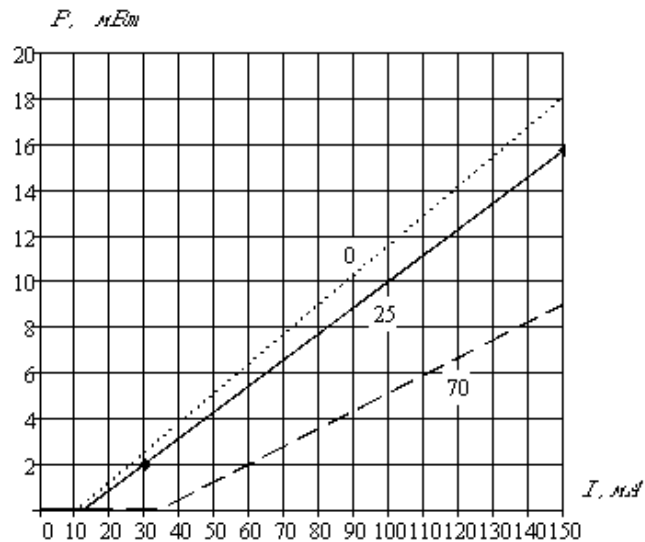
КПД, %	0,05 (паспорт)
--------	-----------------

Пример выполнения (построенные графики)

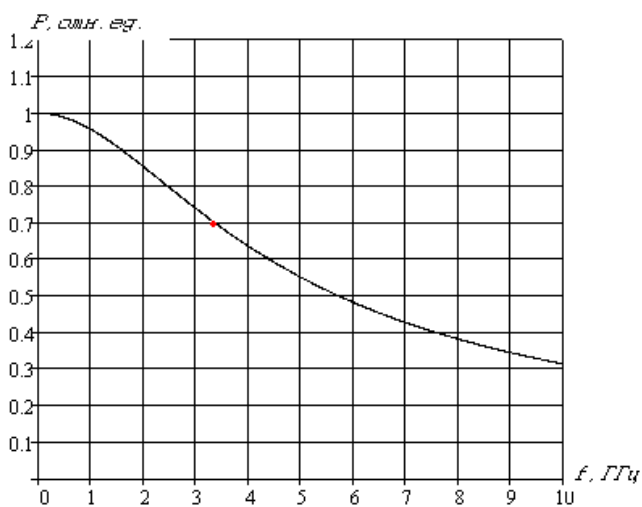
Вольт-амперная характеристика



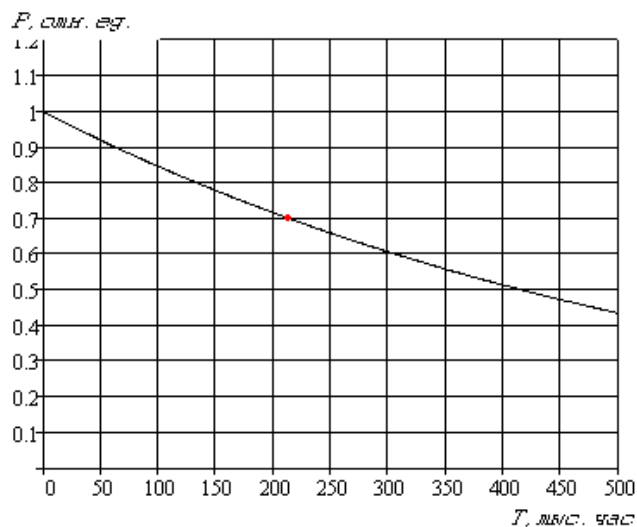
Излучательная характеристика



Частотная характеристика



Характеристика деградации



Тема 2 Компьютерная обработка экспериментальных данных

Для решения задач обработки данных используются различные статистические методы: проверка гипотез, оценивание параметров и числовых характеристик случайных величин и процессов, корреляционный и дисперсионный анализ. В зависимости от точности и сложности эксперимента выбираются и методы обработки данных.

При обработке экспериментальных данных используются следующие понятия:

Функция распределения в теории вероятностей — функция, характеризующая распределение случайной величины или случайного вектора; вероятность того, что случайная величина X примет значение, меньшее или равное x , где x — произвольное действительное число. При соблюдении известных условий полностью определяет случайную величину.

Аппроксимация (от лат. *proxima* — ближайшая) или приближение — научный метод, состоящий в замене одних объектов другими, в каком-то смысле близкими к исходным, но более простыми. Аппроксимация позволяет исследовать числовые характеристики и качественные свойства объекта, сводя задачу к изучению более простых или более удобных объектов (например, таких, характеристики которых легко вычисляются или свойства которых уже известны).

Интерполяция, интерполирование (от лат. *inter-polis* — «разглаженный, подновлённый, обновлённый; преобразованный») — в вычислительной математике способ нахождения промежуточных значений величины по имеющемуся дискретному набору известных значений.

Многим из тех, кто сталкивается с научными и инженерными расчётами, часто приходится оперировать наборами значений, полученных опытным путём или методом случайной выборки. Как правило, на основании этих наборов требуется построить функцию, на которую могли бы с высокой точностью попадать другие

получаемые значения. Такая задача называется аппроксимацией. Интерполяцией называют такую разновидность аппроксимации, при которой кривая построенной функции проходит точно через имеющиеся точки данных.

Существует также близкая к интерполяции задача, которая заключается в аппроксимации какой-либо сложной функции другой, более простой функцией. Если некоторая функция слишком сложна для производительных вычислений, можно попытаться вычислить её значение в нескольких точках, а по ним построить, то есть интерполировать, более простую функцию. Разумеется, использование упрощенной функции не позволяет получить такие же точные результаты, какие давала бы первоначальная функция. Но в некоторых классах задач достигнутый выигрыш в простоте и скорости вычислений может перевесить получаемую погрешность в результатах.

Простейшим способом интерполяции является интерполяция методом ближайшего соседа.

Задание на практическую работу

Самостоятельно провести компьютерную обработку массива экспериментальных данных с применением функций интерполяции и регрессии.

Тема 3 Работа с графическими данными

Важнейшая функция компьютера - обработка информации. Особо можно выделить обработку информации, связанную с изображениями. Она разделяется на три основные направления: компьютерная графика (КГ), обработка и распознавание изображений.

Задача компьютерной графики (Computer Graphics) - визуализация, то есть создание изображения. Визуализация выполняется, исходя из описания (модели) того, что нужно отображать. Существует много методов и алгоритмов визуализации, которые различаются между собою в зависимости от того что и как отображать. Например, отображение того, что может быть только в воображении человека — график функций, диаграмма, схема, карта. Или наоборот, имитация трехмерной реальности — изображение сцен в компьютерных играх, художественных фильмах, тренажерах, в системах архитектурного проектирования. Важными и связанными между собою факторами здесь являются: скорость изменения кадров, насыщенность сцены объектами, качество изображения, учет особенностей графического устройства.

Компьютерная графика (также машинная графика) — область деятельности, в которой компьютеры наряду со специальным программным обеспечением используются в качестве инструмента, как для создания (синтеза) и редактирования изображений, так и для оцифровки визуальной информации, полученной из реального мира, с целью дальнейшей её обработки и хранения.

Обработка изображений (Computer Vision) — это преобразования изображений. Входными данными является изображение, и результат обработки — тоже изображение. Примерами обработки изображений могут служить: повышение контраста, чёткости, коррекция цветов, редукция цветов, сглаживание, уменьшение шумов и так далее. В качестве материала для обработки могут использоваться космические снимки, сканированные изображения, радиолокационные,

инфракрасные изображения и т. п. Задачей обработки изображений может быть как улучшение в зависимости от определенного критерия (реставрация, восстановление), так и специальное преобразование, кардинально меняющее изображения. В последнем случае обработка изображений может быть промежуточным этапом для дальнейшего распознавания изображения.

Распространение компьютерной графики началось с полиграфии. Но вскоре она вырвалась из тесных помещений типографий на простор широкого применения. Огромную популярность завоевали компьютерные игры, научная графика и фильмы. Сейчас без развитой и изощренной графики не обходится ни один фантастический фильм, ни одна компьютерная игрушка. Создаются изображения настолько реальные, что трудно поверить в то, что все это создано на компьютере. Мощнейшие машины и талантливейшие команды математиков, программистов и дизайнеров работают над этим. Ни один приличный доклад в сфере бизнеса не обходится сейчас без компьютерной презентации.

Из простого перечисления областей применения видно, что понятие компьютерной графики довольно обширно — от алгоритмов, рисующих на экране причудливые узоры, до мощных пакетов 3D-графики и программ, имитирующих классические инструменты художника. Иными словами, компьютерная графика не является простым рисованием при помощи компьютера, а представляет собой довольно сложный комплекс, который находит применение во многих областях человеческой деятельности:

Научная графика — первые компьютеры использовались лишь для решения научных и производственных задач. Чтобы лучше понять полученные результаты, производили их графическую обработку, строили графики, диаграммы, чертежи рассчитанных конструкций. Первые графики на машине получали в режиме символьной печати. Затем появились специальные устройства — графопостроители (плоттеры) для вычерчивания чертежей и графиков чернильным пером на бумаге. Современная научная компьютерная графика дает возможность проводить вычислительные эксперименты с наглядным представлением их результатов.

Деловая графика — область компьютерной графики, предназначенная для наглядного представления различных показателей работы учреждений. Плановые показатели, отчётная документация, статистические сводки — вот объекты, для которых с помощью деловой графики создаются иллюстративные материалы. Программные средства деловой графики включаются в состав электронных таблиц.

Конструкторская графика используется в работе инженеров-конструкторов, архитекторов, изобретателей новой техники. Этот вид компьютерной графики является обязательным элементом САПР (систем автоматизации проектирования). Средствами конструкторской графики можно получать как плоские изображения (проекции, сечения), так и пространственные трёхмерные изображения.

Иллюстративная графика — это произвольное рисование и черчение на экране монитора. Пакеты иллюстративной графики относятся к прикладному программному обеспечению общего назначения. Простейшие программные средства иллюстративной графики называются графическими редакторами.

Художественная и рекламная графика — ставшая популярной во многом благодаря телевидению. С помощью компьютера создаются рекламные ролики, мультфильмы, компьютерные игры, видеоуроки, видеопрезентации. Графические пакеты для этих целей требуют больших ресурсов компьютера по быстродействию и памяти. Отличительной особенностью этих графических пакетов является возможность создания реалистических изображений и «движущихся картинок». Получение рисунков трёхмерных объектов, их повороты, приближения, удаления, деформации связано с большим объёмом вычислений. Передача освещённости объекта в зависимости от положения источника света, от расположения теней, от фактуры поверхности, требует расчётов, учитывающих законы оптики.

Компьютерная анимация — это получение движущихся изображений на экране дисплея. Художник создает на экране рисунки начального и конечного положения движущихся объектов, все промежуточные состояния рассчитывает и изображает компьютер, выполняя расчёты, опирающиеся на математическое

описание данного вида движения. Полученные рисунки, выводимые последовательно на экран с определённой частотой, создают иллюзию движения.

Мультимедиа — это объединение высококачественного изображения на экране компьютера со звуковым сопровождением. Наибольшее распространение системы мультимедиа получили в области обучения, рекламы, развлечений.

Сферы применения компьютерной графики чрезвычайно разнообразны. Каждый ее раздел имеет свои отличительные особенности и тонкости «технологического производства». Для каждого из них создано свое программное обеспечение, включающее разнообразные специальные программы (графические редакторы). Вне зависимости от области использования каждый графический редактор, как правило, должен иметь:

- инструменты рисования на компьютер;
- библиотеку готовых изображений;
- набор шрифтов;
- набор спецэффектов;
- а также быть совместимым с другими графическими программами.

Это программное обеспечение по принципу действия и функциональному назначению можно разделить на 3 группы:

- растровая графика
- векторная графика
- фрактальная графика

Растровая графика всегда оперирует двумерным массивом (матрицей) пикселей. Каждому пикселю сопоставляется значение яркости, цвета, прозрачности — или комбинация этих значений. Растровый образ имеет некоторое число строк и столбцов.

Без особых потерь растровые изображения можно только лишь уменьшать, хотя некоторые детали изображения тогда исчезнут навсегда, что иначе в векторном представлении. Увеличение же растровых изображений оборачивается видом на увеличенные квадраты того или иного цвета, которые раньше были пикселями.

В растровом виде представимо любое изображение, однако этот способ хранения имеет свои недостатки: большой объём памяти, необходимый для работы с изображениями, потери при редактировании.

Векторная графика представляет изображение как набор геометрических примитивов. Обычно в качестве них выбираются точки, прямые, окружности, прямоугольники, а также как общий случай, кривые некоторого порядка. Объектам присваиваются некоторые атрибуты, например, толщина линий, цвет заполнения. Рисунок хранится как набор координат, векторов и других чисел, характеризующих набор примитивов. При воспроизведении перекрывающихся объектов имеет значение их порядок.

Изображение в векторном формате даёт простор для редактирования. Изображение может без потерь масштабироваться, поворачиваться, деформироваться, также имитация трёхмерности в векторной графике проще, чем в растровой. Дело в том, что каждое такое преобразование фактически выполняется так: старое изображение (или фрагмент) стирается, и вместо него строится новое. Математическое описание векторного рисунка остаётся прежним, изменяются только значения некоторых переменных, например, коэффициентов.

При преобразовании растровой картинки исходными данными является только описание набора пикселей, поэтому возникает проблема замены меньшего числа пикселей на большее (при увеличении), или большего на меньшее (при уменьшении). Простейшим способом является замена одного пикселя несколькими того же цвета (метод копирования ближайшего пикселя: Nearest Neighbour). Более совершенные методы используют алгоритмы интерполяции, при которых новые пиксели получают некоторый цвет, код которого вычисляется на основе кодов цветов соседних пикселей. Подобным образом выполняется масштабирование в программе Adobe Photoshop (билинейная и бикубическая интерполяция).

Вместе с тем, не всякое изображение можно представить как набор из примитивов. Такой способ представления хорош для схем, используется для

масштабируемых шрифтов, деловой графики, очень широко используется для создания мультфильмов и просто роликов разного содержания

Фрактал — объект, отдельные элементы которого наследуют свойства родительских структур. Поскольку более детальное описание элементов меньшего масштаба происходит по простому алгоритму, описать такой объект можно всего лишь несколькими математическими уравнениями. Фракталы позволяют описывать целые классы изображений, для детального описания которых требуется относительно мало памяти. С другой стороны, фракталы слабо применимы к изображениям вне этих классов.

Задание на практическую работу

Разработать схему организации связи и ситуационный чертеж прокладки кабеля на участке сети связи между городами в соответствии с правилами оформления чертежей проводных средств связи.

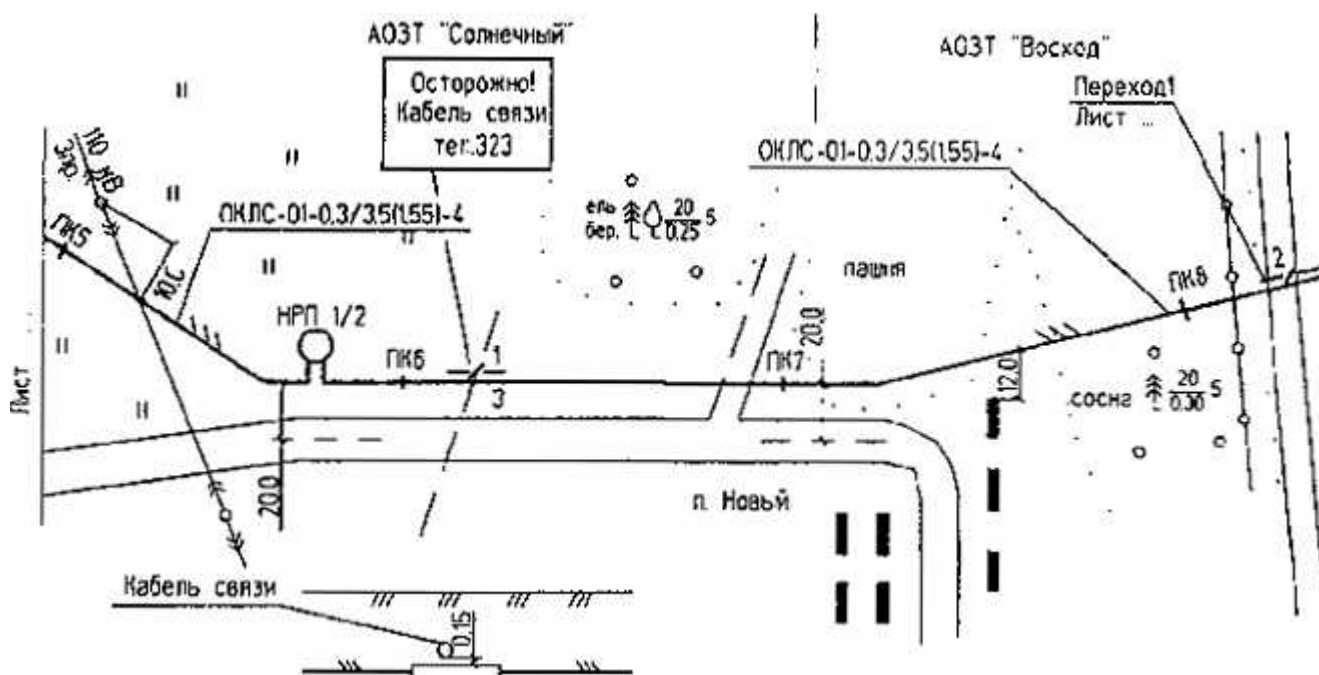


Рисунок 1. Пример выполнения схемы

**Пример выполнения плана и продольного профиля
кабельного перехода через несудоходную реку**

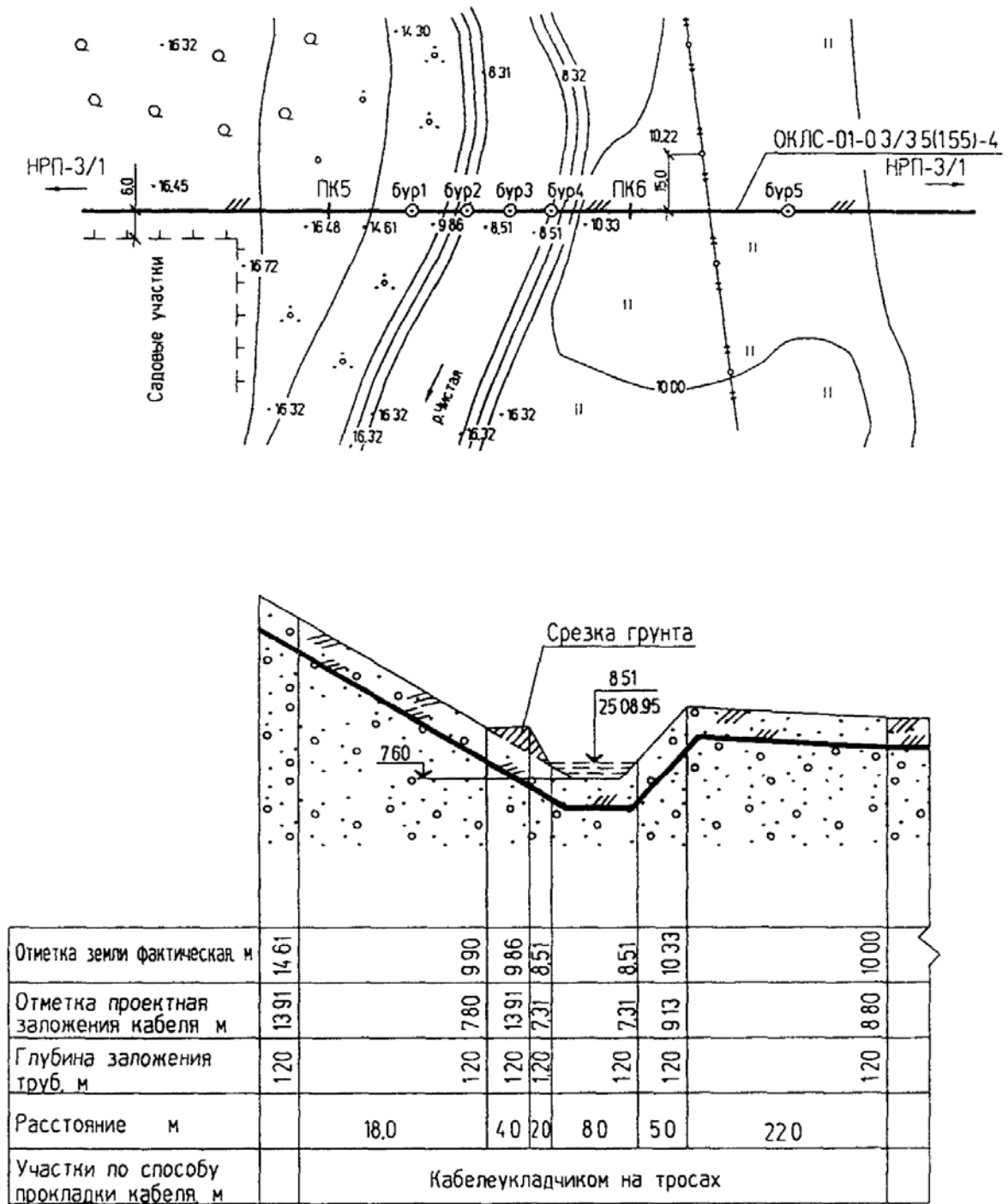


Рисунок 2. Пример выполнения схемы

Тема 4 Средства разработки приложений

На сегодняшний день Qt широко используется разработчиками во всем мире. Библиотека Qt доступна в исходных текстах, сопровождается хорошо проработанной документацией, предоставляет богатые возможности для разработки графического интерфейса. В комплект Qt входит интегрированная среда разработки, включающая редактор кода, интерактивный отладчик, редактор графического интерфейса, справочную систему. Благодаря этому Qt набирает все бóльшую популярность. Пользователями Qt являются многие известные компании, среди которых AT&T, Canon, HP, Bosch, IBM, Motorola, NEC, Sony, Siemens, Sharp, Xerox. Модули Qt использованы при написании известных программ: Skype, Adobe Photoshop Album, Google Earth и др.

Создайте новый проект. Для создания нового проекта нажмите кнопку «Новый проект» и задайте параметры нового проекта. Необходимо создать приложение с графическим интерфейсом пользователя, поэтому в появившемся окне «Новый проект» выберите шаблон «Приложение Qt Widgets» и нажмите кнопку «Выбрать...».

В следующем появившемся окне, задайте название проекта, например «*first*». Укажите путь для размещения проекта (можно оставить по умолчанию). В пути размещения следует использовать только латинские символы. Использование кириллицы при задании пути размещения проекта приводит к ошибке компиляции.

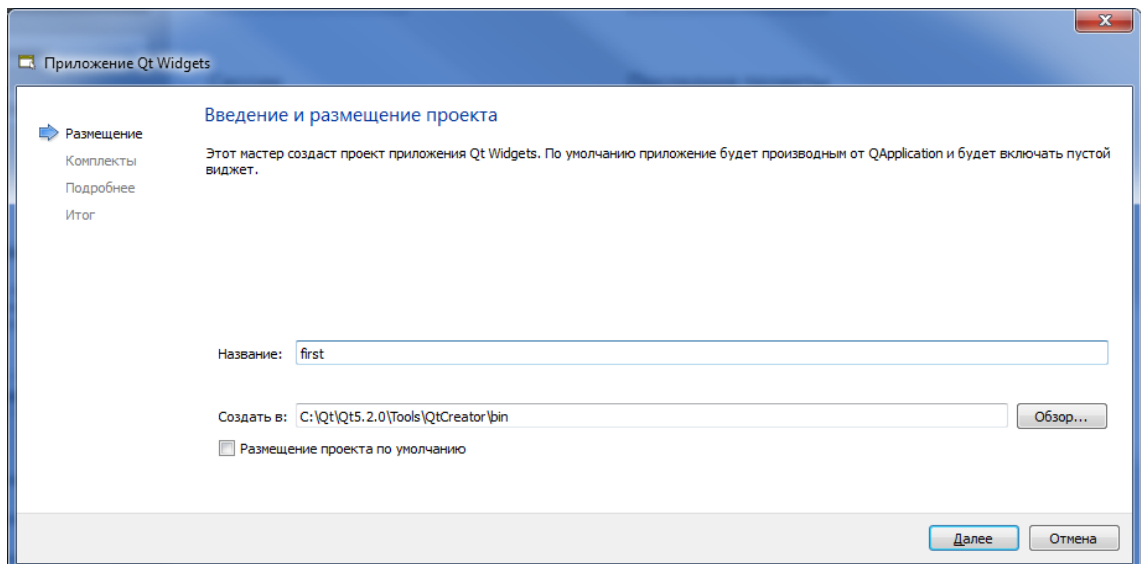


Рис. 1. Задание ввода имени и размещения проекта

Остальные параметры, предлагаемые по умолчанию, оставьте без изменений, несколько раз нажмите кнопку «Далее» и затем нажмите кнопку «Завершить». В результате должно открыться окно среды разработки

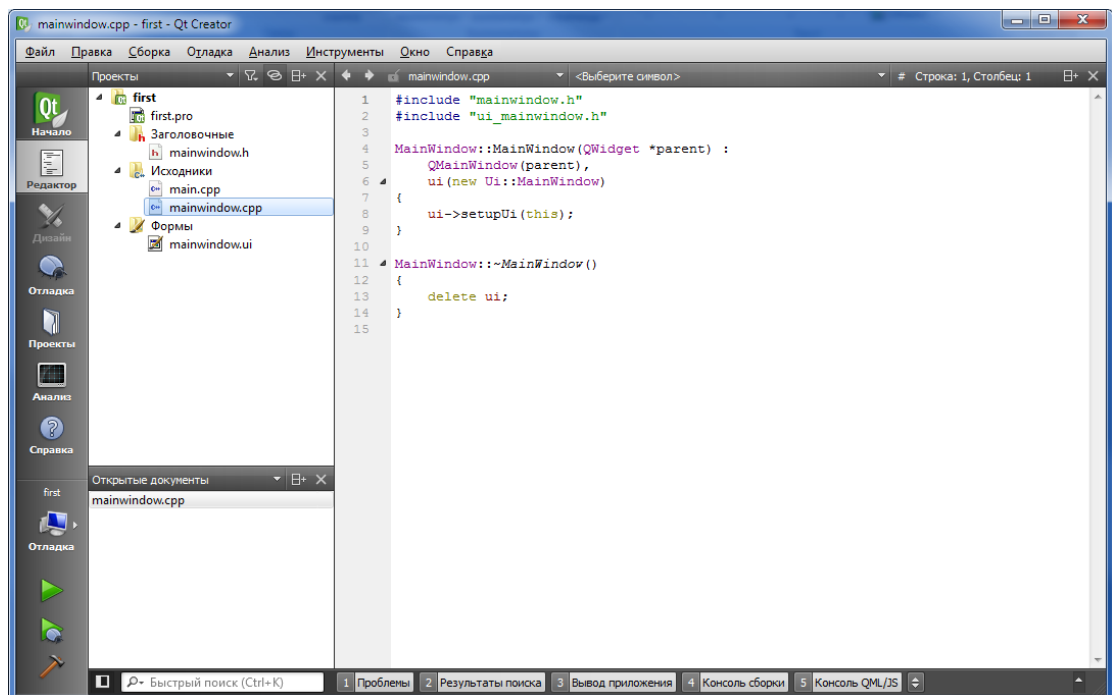


Рис. 2. Окно интегрированной среды разработки. Первый проект

Рассмотрим компоненты графического интерфейса среды разработки:

1) в самом верху расположено главное меню, свойственное многим приложениям Windows;

2) с левой стороны находится полоса смены режимов работы: редактирование, отладка, проекты, справка, а также кнопки компилирования и запуска;

3) правее расположена боковая панель, в которой перечислены файлы, составляющие проект:

файл проекта: `first.pro`;

заголовочный файл: `mainwindow.h`;

исполняемые файлы: `main.cpp`, `mainwindow.cpp`;

файл формы: `mainwindow.ui`.

Если навести мышку на имя файла, то всплывающая подсказка покажет путь, где сохранен файл. Двойной клик мышкой открывает файл для редактирования;

4) самое большое окно среды разработки – окно редактора. В нем можно редактировать код программы;

5) в самом низу располагается окно вывода приложения, в него будут выводиться сообщения о ходе компиляции и сообщения об ошибках (если будут ошибки).

Сохраните проект. Для этого в главном меню выберете команду (Файл) => (Сохранить все).

Запустите созданную программу. Для этого нажмите кнопку с зеленым треугольником.

Сейчас программа представляет собой пустое окно «Main Window». В дальнейшем мы будем добавлять элементы интерфейса и развивать только что созданный проект. Сейчас его можно закрыть. Продолжить работу над проектом можно, запустив Qt Creator и выбрав ссылку «*first*» в списке «Последние проекты».

Итак, мы успешно установили Qt и создали проект (приложение, состоящее из одной формы), запустили его и убедились, что все необходимое установлено правильно.

Усовершенствуем главное окно учебной программы. Для этого разместим на форме элементы управления – виджеты (Widget) и визуальные объекты – кнопки и поля редактирования.

Используя боковую панель, откройте файл `mainwindow.ui` в режиме «Дизайн».

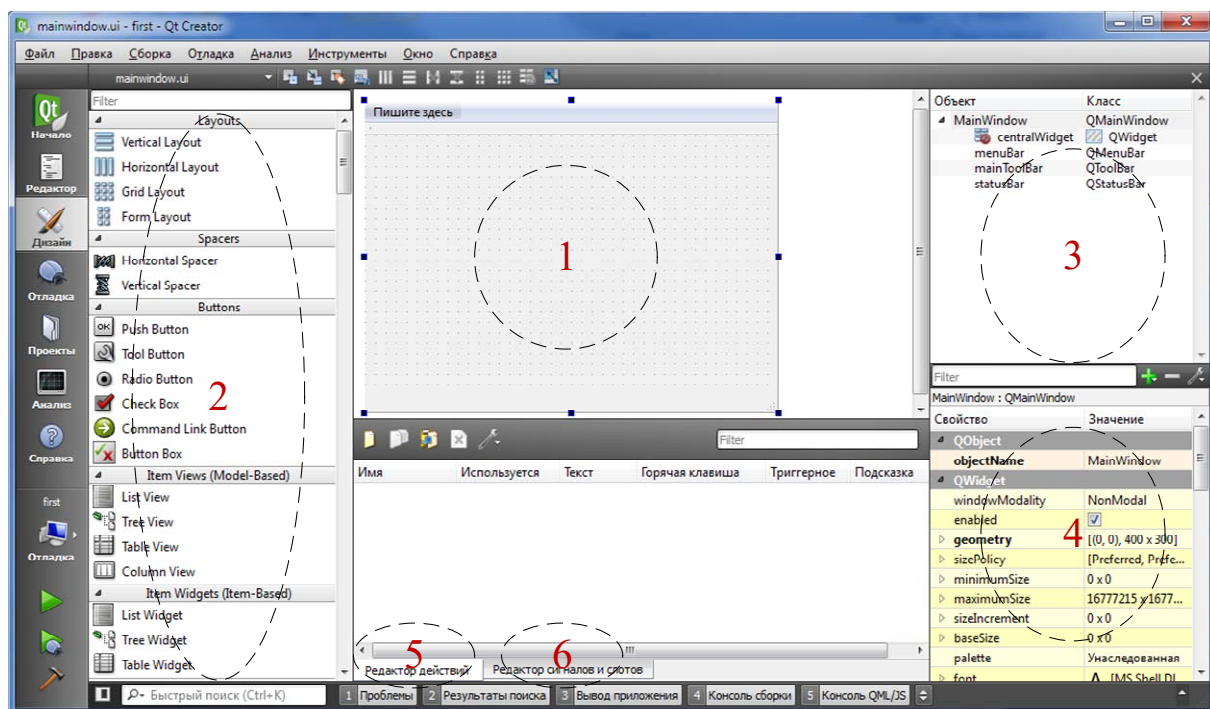


Рис. 3. Разработка интерфейса пользователя

На рис. 3 пунктиром обозначены:

- 1) форма, расположенная в центре окна;
- 2) панель виджетов (Widget Box);
- 3) инспектор объектов (Object Inspector);
- 4) редактор свойств (Property Editor);
- 5) редактор действий;
- 6) редактор сигналов и слотов.

На заготовке формы уже размещены элементы: Меню (`menuBar`) и Панель инструментов (`mainToolBar`). Пусть они остаются без изменений.

Разместите на форме две кнопки (`Push Button`). Для размещения виджетов на форме следует перетащить их мышкой на форму из панели виджетов.

Отредактируйте надписи на кнопках и их имена. Для этого выберите первую из кнопок мышкой на форме (или в инспекторе объектов). В редакторе свойств найдите свойство *text* и измените его значение с *PushButton* на *Очистить*. Измените имя первой кнопки, для этого свойству *objectName* задайте значение *pushButtonClear*.

Аналогично задайте для второй кнопки: свойству *text* – значение *Копировать*, свойству *objectName* – значение *pushButtonCopy*.

Добавьте еще два виджета Line Edit – Поля редактирования. Для первого поля редактирования задайте имя *lineEdit_1*, свойству *text* задайте значение *источник*. Свойству *text* второго поля редактирования с именем *lineEdit_2* задайте значение *получатель*.

После выполнения всех манипуляций окно программы примет вид, аналогичный приведенному на рис. 4.

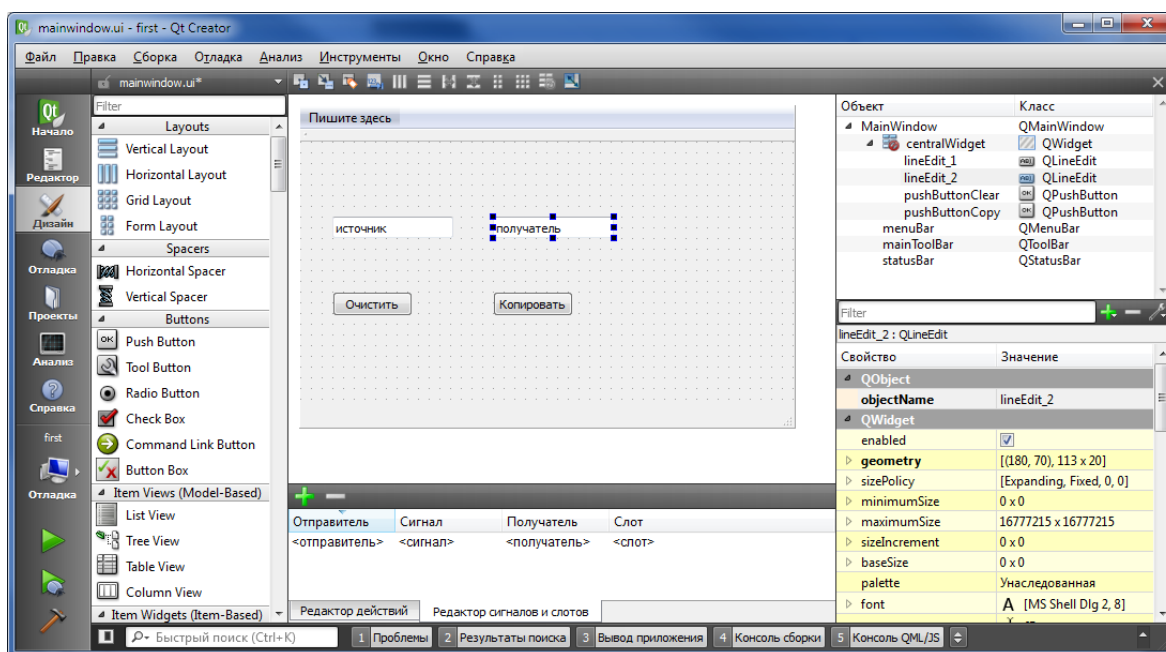


Рис. 4. Разработка интерфейса пользователя

Проверьте правильность выполнения, запустите программу. Перед запуском программы Qt Creator предлагает сохранить изменения в файлах проекта, показывая окно «Сохранение изменений». Сохраните проект, нажав на кнопку «Сохранить все».

Пока программа не выполняет полезных действий, но мы уже познакомились с виджетами, научились изменять их свойства в редакторе свойств и разработали вполне достойный графический интерфейс с кнопками и полями редактирования.

Добавьте на форму элемент Надпись. Для этого из панели виджетов перетащите на форму компонент Label. Переименуйте его. Для этого в редакторе свойств определите свойство *objectName* значением *labelResult*.

Увеличьте элемент Надпись, чтобы в нее влезал выводимый текст. Для этого «схватите» границу надписи мышкой и растяните.

Измените текст надписи из кода программы, открыв файл `mainwindow.cpp`, затем после строки

```
ui->setupUi(this);
```

добавьте строку

```
ui->labelResult->setText("Результат!");
```

При редактировании кода программы среда разработки Qt Creator выполняет автоматическое дополнение кода, предлагая список соответствующих альтернатив, как показано на рис. 5.

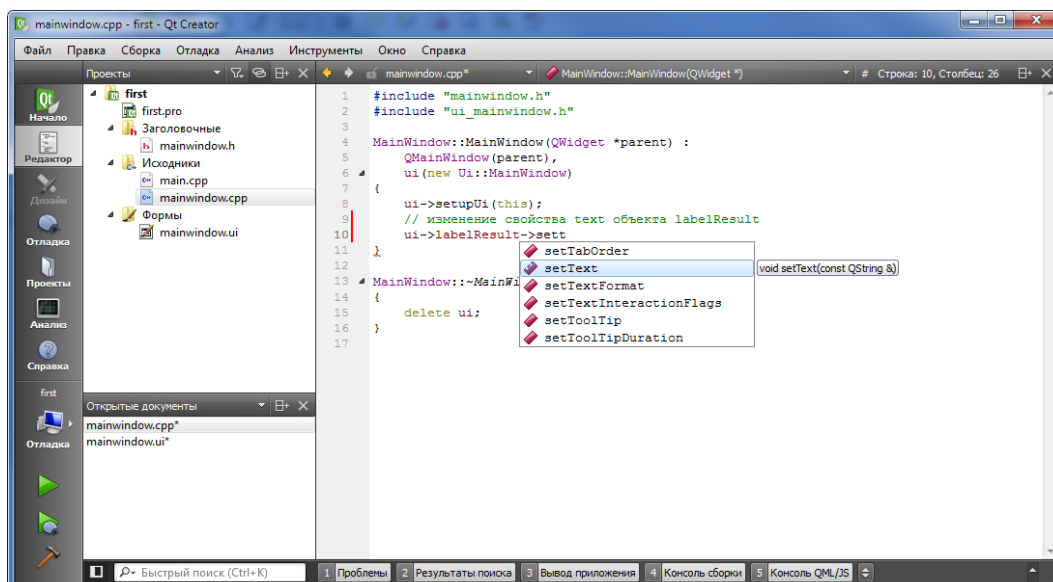


Рис. 5. Автоматическое дополнение кода

Редактор среды разработки выделяет различными цветами синтаксические элементы, что упрощает визуальное восприятие и облегчает обнаружение синтаксических ошибок.

После изменений листинг файла `mainwindow.cpp` должен иметь следующий вид:

```
1:     #include "mainwindow.h"
2:     #include "ui_mainwindow.h"
3:
4:     MainWindow::MainWindow(QWidget *parent) :
5:         QMainWindow(parent),
6:         ui(new Ui::MainWindow)
7:     {
8:         ui->setupUi(this);
9:         // изменение свойства text объекта labelResult
10:        ui->labelResult->setText("Результат!");
11:    }
12:
13:    MainWindow::~MainWindow()
14:    {
15:        delete ui;
16:    }
```

Пояснения к программе

В первых двух строках (1 и 2) подключаются заголовочные файлы модулей.

Далее следует код конструктора `MainWindow`. Код, расположенный между фигурными скобками (строки 7–11), будет выполняться при запуске программы.

Строка 9 – однострочный комментарий. Строки, начинающиеся с двух косых черточек (слэшей), служат для пояснений текста программы и игнорируются компилятором.

Строка 10 – вызов метода (функции), изменяющего свойство *text* элемента *labelResult*. В скобках указан аргумент функции – строка «Результат!». Для корректного обращения к виджетам созданной формы перед их именем следует писать `ui->`

Строки 13–16 связаны с уничтожением окна.

Проверьте корректность внесенных дополнений, запустите программу. Текст надписи на форме «*TextLabel*» изменяется на «*Результат!*» из кода программы.

Итак, мы научились изменять размеры компонентов на форме с помощью мыши, обращаться к свойствам элементов из кода программы и изменять их.

Задание на практическую работу

Самостоятельно разработать компьютерную программу по расчету натяжения подвешенного самонесущего кабеля.

Перечень основной и дополнительной литературы

Основная литература:

1. Информатика. Базовый курс [Текст] : учеб. пособие для вузов / ред. С. В. Симонович ; рец. С. В. Калинин. - СПб. : Питер, 2001. - 639 с. : ил. - Библиогр.: с. 620-622. - Алф. указ.: с. 623-638. - ISBN 5-8046-0134-2 (в пер.) : 107.25 р.
2. Информатика. Базовый курс [Текст] : учеб. пособие для втузов / Под ред. С. В. Симоновича. - СПб. : Питер, 2002. - 640 с. : ил. - (Учебник для вузов). - Библиогр.: с. 620-622. - ISBN 5-8046-0134-2 : 107.25 р.
3. Информатика. Базовый курс [Текст] : учеб. пособие для втузов / Под ред. С. В. Симоновича. - СПб. : Питер, 2003. - 640 с. : ил. - (Учебник для вузов). - Библиогр.: с. 620-622. - ISBN 5-8046-0134-2 : 278.40 р., 107.25 р.
4. Информатика. Базовый курс [Текст] : учеб. пособие для втузов / под ред. С. В. Симоновича. - 2-е изд. - СПб. : Питер, 2005. - 640 с. : ил. - (Учебник для вузов). - Библиогр. : с. 631-632. - ISBN 5-94723-752-2 : 300.00 р.
5. Системное программное обеспечение [Текст] : учеб. пособие для вузов / М. А. Лейкин [и др.] ; рец. Ю. М. Смирнов ; М-во связи и массовых коммуникаций Рос. Федерации, Федер. агентство связи, Гос. образовательное учреждение высш. проф. образования "С.-Петербург. гос. ун-т телекоммуникаций им. проф. М. А. Бонч-Бруевича", М-во образования и науки Рос. Федерации, Воронеж. гос. техн. ун-т, Курский гос. техн. ун-т. - СПб. : СПбГУТ, 2010. - 140 с. : ил. - Библиогр. : с. 138-139. - ISBN 978-5-89160-055-3 : 170.00 р., 400.00 р.
6. Информатика. Базовый курс [Текст] : учеб. пособие для втузов / ред. С. В. Симонович ; рец. С. В. Калинин. - 2-е изд. - СПб. : Питер, 2007. - 639 с. : ил. - (Учебник для вузов). - Библиогр.: с. 631-632. - Алф. указ.: с. 633-639. - ISBN 5-94723-752-0 (в пер.) : 223.00 р., 300.00 р.
7. Красов, А. В. Компьютерное обеспечение инженерных задач [Текст] : метод. указ. к лаб. работам / А. В. Красов, А. С. Верещагин ; рец. С. Е. Душин ; Федер.

- агентство связи, Федер. гос. образовательное бюджет. учреждение высш. проф. образования "С.-Петерб. гос. ун-т телекоммуникаций им. проф. М. А. Бонч-Бруевича". - СПб. : СПбГУТ, 2011. - 23 с. : ил. - Библиогр.: с.19. - (в обл.) : 10.50 р.
8. Информатика. Базовый курс [Текст] : учебник / ред. С. В. Симонович ; рец. С. В. Калинин. - 2-е изд. - СПб. : Питер, 2009. - 639 с. : ил. - ISBN 978-5-94723-752-8 : 300.00 р.
9. Информатика. Базовый курс [Текст] : учебник для вузов / ред. С. В. Симонович. - 3-е изд. - СПб. : Питер, 2011. - 637 с. : ил. - (Учебник для вузов). - ISBN 978-5-4-9-00439-7 : 400.00 р.
10. Веретехина, С. В. Информационные технологии. Пакеты программного обеспечения общего блока «IT-инструментарий» [Электронный ресурс] : учебное пособие / Веретехина С. В. - Москва : Русайнс, 2015. - 44 с. - ISBN 978-5-43-5-0177-2 Книга находится в Премиум-версии ЭБС IPRbooks.
11. Хрящев, В. Моделирование и создание чертежей в системе AutoCAD [Электронный ресурс] / В. Хрящев, Г. Шипова. - Санкт-Петербург : БХВ-Петербург, 2015. - 224 с. : ил. - ISBN 978-5-9775-2001-0 : Б. ц.
12. Павловская, Т. А. C/C++. Процедурное и объектно-ориентированное программирование. Учебник для вузов. Стандарт 3-го поколения [Электронный ресурс] / Т.А. Павловская. - Санкт-Петербург : Питер, 2015. - 496 с. : ил. - ISBN 978-5-496-00109-0 : Б. ц.
13. Информатика. Базовый курс [Текст] : учебник / ред. С. В. Симонович ; рец. С. В. Калинин. - 2-е изд. - М. [и др.] : Питер, 2005. - 640 с. : ил. - (Учебник для вузов). - Библиогр.: с. 631-632. - Алф. указ.: с. 633-639. - ISBN 5-94723-752-0 : 300.00 р.
14. Информатика. Базовый курс [Текст] : учебник / ред. С. В. Симонович ; рец. С. В. Калинин. - 2-е изд. - М. [и др.] : Питер, 2004. - 640 с. : ил. - (Учебник для вузов). - Библиогр.: с. 631-632. - Алф. указ.: с. 633-639. - ISBN 5-94723-752- 0 : 300.00 р.

Дополнительная литература:

1. Романец, Юрий Васильевич. Защита информации в компьютерных системах и сетях [Текст] : производственно-практическое издание / Ю. В. Романец, П. А. Тимофеев, В. Ф. Шаньгин ; ред. В. Ф. Шаньгин. - 2-е изд., перераб. и доп. - М. : Радио и связь, 2001. - 376 с. : ил. - ISBN 5-256-01518-4 : 70.00 р., 175.05 р.
2. Крерафт, Д. Аналоговая электроника. Схемы, системы, обработка сигнала [Текст] : пер. с англ. / Д. Крерафт, С. Джерджи ; пер. А. А. Кузьмичева ; ред. А. А. Лапин. - М. : Техносфера, 2005. - 360 с. : ил. - (Мир электроники). VII<>14). - Библиогр. : с. 358-359. - ISBN 5-94836-057-1. - ISBN 0-7506-5095-8 : 225.69 р.
3. Бузюков, Л. Б. Современные методы программирования на языках С и С++ [Текст] : учеб. пособие / Л. Б. Бузюков, О. Б. Петрова ; рец.: Э. А. Акчурин, А. Р. Лисс. - СПб. : Линк, 2008. - 287 с. : ил. - Библиогр.: с. 286-287. - ISBN 978-5-985-2-013-7 (в пер.) : 293.70 р.
4. Воронцова, Ирина Олеговна. Программирование на языке высокого уровня С/С++ [Текст] : учеб. пособие / И. О. Воронцова, Л. А. Груздева, Т. В. Губанова ; рец. А. И. Солонина ; Федер. агентство связи, Гос. образовательное учреждение высш. проф. образования "С.-Петерб. гос. ун-т телекоммуникаций им. проф. М. А. Бонч-Бруевича". - СПб. : СПбГУТ, 2010. - 111 с. : ил. - Библиогр.: с. 111. - (в обл.) : 109.16 р.
5. Большаков, В. П. Основы 3D-моделирования. Изучаем работу в AutoCAD, КОМПАС-3D, SolidWorks, Inventor [Текст] : учебный курс / В. П. Большаков, А. Л. Бочков ; рец. Л. А. Голдобина. - СПб. : Питер, 2013. - 303 с. : ил. - Библиогр.: с. 300. - ISBN 978-5-496-00041-3 (в обл.) : 352.50 р.
6. Биллиг, В. А. Основы объектного программирования на С# (С# 3.0, Visual Studio 2008) [Электронный ресурс] : учебное пособие / Биллиг В. А. - Москва : БИНОМ. Лаборатория знаний, Интернет-Университет Информационных Технологий (ИНТУИТ), 2010. - 582 с. - ISBN 978-5-9963-0259-8 : Б. ц. Книга находится в Премиум-версии ЭБС IPRbooks.
7. Латышев, П. Н. Каталог САПР [Текст] : программы и производители. 2014-2015 /

- Латышев П. Н. - Москва : СОЛОН-ПРЕСС, 2013. - 694 с. - ISBN 978-5-91359-142-5 :
Б. ц. Книга находится в Премиум-версии ЭБС IPRbooks.
8. Кирьянов, Д. Самоучитель Mathcad 11 [Электронный ресурс] / Д. Кирьянов. -
Санкт-Петербург : БХВ-Петербург, 2014. - 560 с. : ил. - ISBN 978-5-9775-1977-9 :
Б. ц.
9. Русинович, М. Внутреннее устройство Microsoft Windows. 6-е изд. Основные
подсистемы ОС [Электронный ресурс] / М. Русинович, Д. . Соломон, Алекс
Ионеску. - Санкт-Петербург : Питер, 2014. - 672 с. : ил. - ISBN 978-5-496-00791-7 :
Б. ц.
10. Шлее, М. Qt 5.3. Профессиональное программирование на C++ [Электронный
ресурс] / М. Шлее. - Санкт-Петербург : БХВ-Петербург, 2015. - 928 с. : ил. - ISBN
978-5-9775-3346-1 : Б. ц.
11. Иванова, Н. Ю. Системное и прикладное программное обеспечение
[Электронный ресурс] : учебное пособие / Иванова Н. Ю. - Москва : Прометей,
2011. - 202 с. - ISBN 978-5-4263-0078-1 : Б. ц. Книга находится в Премиум-версии
ЭБС IPRbooks.
12. Дюбов, Андрей Сергеевич. Компьютерное обеспечение расчетно-проектной и
экспериментально-исследовательской деятельности [Текст] : учебно-методическое
пособие по выполнению курсовой работы / А. С. Дюбов ; рец. И. В.
Гришин ; Федер. агентство связи, Федеральное государственное бюджетное
образовательное учреждение высшего образования "Санкт-Петербургский
государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича". -
СПб. : СПбГУТ, 2017. - 31 с. : ил. - 493.42 р.