



Федеральное государственное образовательное бюджетное учреждение высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)

Кафедра информационных технологий
Факультет информационных технологий и анализа больших данных

Документ подписан усиленной неквалифицированной электронной подписью.

Организация: «ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ»

Сертификат: VZNaL5FyvBclcFH6lLm8bGEZK/fdYlрT

Утверждено: Проректор по учебной и методической работе, Е.А. Каменева

Дата: 10.02.2026 г.

О.Н. Крахмалев
Программирование для встраиваемых систем

Рабочая программа дисциплины

для студентов, обучающихся по направлению подготовки
01.03.02 - Прикладная математика и информатика,
Образовательная программа «Прикладное машинное обучение»
Профиль:
«Прикладное машинное обучение»

Рекомендовано

*Факультет информационных технологий и анализа больших данных
(протокол № 08 от 19.05.2026 г.)*

Одобрено

*Кафедра информационных технологий
(протокол № 03 от 05.05.2026 г.)*



Содержание

1. Наименование дисциплины	3
2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине	3
3. Место дисциплины в структуре образовательной программы	
4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся	4
5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий	5
5.1. Содержание дисциплины	5
5.2. Учебно – тематический план	7
5.3. Содержание семинаров, практических занятий	8
6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	10
6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы	10
6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю	11
7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине	12
8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	18
9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	18
10. Методические указания для обучающихся по освоению дисциплины	22
11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)	24
12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	24



1. Наименование дисциплины

Программирование для встраиваемых систем

2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
ПКП-7	Способность создавать прикладные программные компоненты и интегрировать в них методы анализа данных, машинного обучения и искусственного интеллекта	Владеет инструментальными средствами разработки прикладных программных продуктов: языками программирования, библиотеками, фреймворками	Знать: средства разработки прикладных программных продуктов: языки программирования, библиотеки, фреймворки Уметь: разрабатывать программные продукты с использованием библиотек и фреймворков
		Владеет архитектурными принципами и паттернами создания прикладных программных продуктов на различных платформах	Знать: принципы и паттерны создания прикладных программных продуктов Уметь: создавать прикладные программные продукты на различных платформах
		Демонстрирует понимание побочных процессов, сопровождающих прикладную разработку программного обеспечения в промышленных условиях и владеет соответствующим инструментарием	Знать: побочные процессы, сопровождающих прикладную разработку программного обеспечения в промышленных условиях Уметь: использовать соответствующий инструментарий для сопровождения прикладного программного обеспечения



3. Место дисциплины в структуре образовательной программы

Дисциплина «Программирование для встраиваемых систем» относится к «Модулю "Системное программирование"».

4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

Очная форма обучения

Вид учебной работы по дисциплине	Всего (в з/е и часах)	Семестр 7 (в часах)
Общая трудоемкость дисциплины	3/108	108
<i>Контактная работа – Аудиторные занятия</i>	<i>50</i>	<i>50</i>
<i>Лекции</i>	<i>16</i>	<i>16</i>
<i>Семинары, практические занятия</i>	<i>34</i>	<i>34</i>
<i>Самостоятельная работа</i>	<i>58</i>	<i>58</i>
Вид текущего контроля	Контрольная работа	Контрольная работа
Вид промежуточной аттестации	Зачет	Зачет



5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Основные понятия, характеристики встраиваемых систем

Основные понятия, классификация встраиваемых систем. Аналоговый и цифровой сигналы. Электронное устройство. Искажение сигналов. Преимущества и недостатки цифровых сигналов. Виды цифровых сигналов. Типы цифровых устройств. Три модели цифровых устройств. Три типа выходов. Соединение выходов разных типов. Типы организации связей. Цифровые элементы, узлы, микросхемы. Понятия двоичной логики. Обзор современных встраиваемых систем. Основные проблемы разработки встраиваемых систем. Особенности и ограничения программных средств встраиваемых систем.

Тема 2. Программирование встраиваемых систем

Командный интерпретатор, права доступа и процессы. Общие принципы и примеры составления компьютерных программ для встроженных систем. Обзор инструментальных средств для программирования встроженных систем. Среда разработчика, библиотеки, трансляторы, редакторы связей, отладка. Средства программирования встраиваемых систем. Обзор средств разработки ПО для встраиваемых систем. Инструментальные среды разработки для микроконтроллеров семейства AVR. Элементы языка программирования C и C++. Примеры программ для AVR и Arduino.

Тема 3. Методология разработки программного обеспечения

Основы программирования на языке Си. Архитектура RISC. Структура программы для микроконтроллера. Порты ввода вывода. Прерывания. Таймеры. Использование широтноимпульсной модуляции. Работа с памятью EEPROM, FLASH. Организация связи с персональным компьютером (интерфейс RS-232). Эмуляторы и логические анализаторы. Эффективные шаблоны доступа к различным типам памяти микроконтроллера. Подсистема аналого-цифрового преобразования микроконтроллера. Устройства дискретного ввода: кнопки, переключатели, клавиатуры. Устройства индикации: светодиоды, семисегментные индикаторы, индикаторы логического выхода с тремя состояниями. Изучение интерфейса UART.



Изучение интерфейса I2C и работа с цифровыми датчиками. Изучение интерфейса SPI и работа с цифровыми модулями.

Тема 4. Операционные системы реального времени для встраиваемых систем

Операционные системы реального времени для встраиваемых систем. Особенности встраиваемых систем на базе Linux. Отличия «встраиваемых» Linux-систем и систем на базе Android от Desktop-версий. Построение ядра встраиваемой операционной системы.

Тема 5. Основы программирования на Arduino

Объяснение основ программирования под Arduino. Средства программирования Arduino. Обучение языку программирования. Режимы обмена информацией с периферийными устройствами (ПУ). Программный ввод-вывод, ввод-вывод по прерываниям, прямой доступ к оперативной памяти. Изучение основных типов данных. Изучение основных функций. Изучение остальных основных компонент синтаксиса языка. Рассмотрение синтаксиса языка на конкретных примерах. Работа в режимах прерывания и захвата. Рассмотрение некоторых методов принятия решения.

Тема 6. Инструментальные программные и аппаратные модули National Instruments

Инструментальные программные и аппаратные модули National Instruments. Аппаратные платформы National Instruments: NI CompactRIO, программируемый промышленный контроллер на базе ПЛИС; NI roboRIO, роботизированный контроллер, используемый в стандарте FIRST Robotics Competition NI CompactDAQ, системы сбора данных для USB и Ethernet. Модельное проектирование встраиваемых систем в LabVIEW.



5.2. Учебно – тематический план

Очная форма обучения

п/ п	Наименование тем (разделов) дисциплины	Трудоемкость в часах				
		Всего	Контактная работа - Аудиторная работа			Самостоятельная работа
			Общая, в т.ч.:	Лекции	Семинары, практические занятия	
1	Основные понятия, характеристики встраиваемых систем	14	6	2	4	8
2	Программирование встраиваемых систем	18	10	4	6	8
3	Методология разработки программного обеспечения	22	12	6	6	10
4	Операционные системы реального времени для встраиваемых систем	18	8	2	6	10
5	Основы программирования на Arduino	22	8	2	6	14
6	Инструментальные программные и аппаратные модули National Instruments	14	6	0	6	8
В целом по дисциплине		108	50	16	34	58



5.3. Содержание семинаров, практических занятий

Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарах, практических занятиях	Формы проведения занятий
Основные понятия, характеристики встраиваемых систем	Что такое встраиваемая система. Основные компоненты встраиваемой системы. Применение встраиваемых систем в различных областях (промышленности, связи, быту, автомобилях, медицине).	Занятия в интерактивной форме в виде дискуссий.
Программирование встраиваемых систем	Стандартные типы данных, операторы, структура программы на языке C и C++. Функции, библиотеки функций. Разработка приложения на языке C. Низкоуровневая оптимизация разработанного приложения.	Занятия в интерактивной форме в виде дискуссий.
Методология разработки программного обеспечения	Работа с портами ввода-вывода. Организация вывода информации. Организация ввода информации. Основы цифровой и аналоговой схемотехники. Аналого-цифровое и цифро-аналоговое преобразования. Классификация периферийных устройств. Работа с микроконтроллером. Таймеры. Прерывания. Использование реле. Широтно-импульсная модуляция. Подключение светодиода через сопротивление.	Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений.
Операционные системы реального времени для встраиваемых систем	Особенности встраиваемых систем на базе Linux. Построение ядра встраиваемой операционной системы. Программное обеспечение для построения и развертывания образа встраиваемой системы.	Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений.



Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарах, практических занятиях	Формы проведения занятий
Основы программирования на Arduino	Основы программирования микроконтроллеров, характеристики и применение микроконтроллеров, изучение Arduino IDE. Управление светодиодами средствами виртуальной среды. Autodesk Circuits с применением условного оператора if. Управление RGB-светодиодом средствами виртуальной среды Autodesk Circuits» с применением цикла for.	Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений.
Инструментальные программные и аппаратные модули National Instruments	Модельное проектирование встраиваемых систем в LabVIEW.	Занятия в интерактивной форме в виде дискуссий.



6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
Основные понятия, характеристики встраиваемых систем	Искажение сигналов. Понятия двоичной логики.	Работа с учебной литературой. Разбор вопросов по теме занятия.
Программирование встраиваемых систем	Примеры программ для AVR и Arduino.	Работа с учебной литературой. Разбор вопросов по теме занятия. Выполнение домашней контрольной работы.
Методология разработки программного обеспечения	Эмуляторы и логические анализаторы. Эффективные шаблоны доступа к различным типам памяти микроконтроллера.	Работа с учебной литературой. Разбор вопросов по теме занятия.
Операционные системы реального времени для встраиваемых систем	Построение ядра встраиваемой операционной системы.	Работа с учебной литературой. Разбор вопросов по теме занятия.
Основы программирования на Arduino	Работа в режимах прерывания и захвата.	Работа с учебной литературой. Разбор вопросов по теме занятия.
Инструментальные программные и аппаратные модули National Instruments	Роботизированный контроллер, используемый в стандарте FIRST Robotics Competition NI CompactDAQ.	Работа с учебной литературой. Разбор вопросов по теме занятия.



6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю

Примерная тематика контрольной работы

1. Разработка и тестовое моделирование на Arduino системы контроля допуска в здание.
2. Разработка и тестовое моделирование на Arduino калькулятора.
3. Разработка и тестовое моделирование на Arduino часов.
4. Разработка и тестовое моделирование на Arduino цифровой клавиатуры для ПК.
5. Разработка и тестовое моделирование на Arduino системы пульта управления.

Дополнительная информация

Критерии балльной оценки различных форм текущего контроля успеваемости содержатся в соответствующих методических рекомендациях Кафедры информационных технологий Факультета информационных технологий и анализа больших данных.



7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Перечень компетенций с указанием индикаторов их достижения в процессе освоения образовательной программы содержится в разделе 2. «Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине».

Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, умений и знаний

Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
ПКП-7. Способность создавать прикладные программные компоненты и интегрировать в них методы анализа данных, машинного обучения и искусствен-	Владеет инструментальными средствами разработки прикладных программных продуктов: языками программирования, библиотеками, фреймворками	Знать: средства разработки прикладных программных продуктов: языки программирования, библиотеки, фреймворки Уметь: разрабатывать программные продукты с использованием библиотек и фреймворков	1. Составьте программу с использованием стандартных библиотек на языке Python. 2. Составьте программу с использованием стандартных библиотек на языке C++.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
ного интеллекта	Владеет архитектурными принципами и паттернами создания прикладных программных продуктов на различных платформах	Знать: принципы и паттерны создания прикладных программных продуктов Уметь: создавать прикладные программные продукты на различных платформах	1. Создать прикладной программный продукт на платформе языка Python. 2. Создать прикладной программный продукт на платформе языка C++.
	Демонстрирует понимание побочных процессов, сопровождающих прикладную разработку программного обеспечения в промышленных условиях и владеет соответствующим инструментарием	Знать: побочные процессы, сопровождающих прикладную разработку программного обеспечения в промышленных условиях Уметь: использовать соответствующий инструментарий для сопровождения при-	1. Создать прикладной программный продукт с интерфейсом сопровождения прикладного программного обеспечения на платформе языка Python. 2. Создать прикладной программный продукт с интерфейсом сопровождения прикладного программного обеспечения на платформе языка C++.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
		кладного программного обеспечения	



Примеры практико-ориентированных заданий

Двухосевой трекер слежения за солнцем

Двухосевая система слежения за солнцем, или регулируемая солнечная панель, это автоматизированная система, которая регулирует ориентацию солнечной панели по двум осям вращения: азимуту (горизонтальному) и углу места (вертикальному), позволяя ей следовать за солнцем в течение светового дня.

Компоненты	Количество	Назначение
Arduino UNO	1	Главный контроллер системы слежения
LDR (5 мм)	4	Светозависимые резисторы определяют направление солнечного света
Панель	1	Улавливает солнечную энергию
Серводвигатель	2	Управляет движением по горизонтальной и вертикальной осям
Резистор на 10 кОм	4	Формирует схему делителя напряжения с фоторезисторами
Макетная доска	1	Прототипирование схем и соединений
Соединительные кабели - компонентами		Электрические соединения между компонентами

Программный код:

```
#include <Servo.h>

// ===== Servo Setup =====

Servo horizontal; // Horizontal Servo Motor
Servo vertical; // Vertical Servo Motor
int servohori = 180; // Initial horizontal angle
int servovert = 45; // Initial vertical angle
// Servo angle limits
int servohoriLimitHigh = 175;
int servohoriLimitLow = 5;
int servovertLimitHigh = 100;
int servovertLimitLow = 1;
// ===== LDR Pins =====
int ldrlt = A0; // Top Left LDR
int ldrrt = A3; // Top Right LDR
int ldrlb = A1; // Bottom Left LDR
```



```
int ldrrd = A2; // Bottom Right LDR
// ===== Function: Read Average LDR Value =====
int readAverage(int pin) {
    long total = 0;
    for (int i = 0; i < 10; i++) {
        total += analogRead(pin);
        delay(2); // small sampling delay for stable reading
    }
    return total / 10;
}

void setup() {
    Serial.begin(9600);
    Serial.println(-----);
    Serial.println("Dual Axis Solar Tracker System Started");
    Serial.println(-----);
    horizontal.attach(2); // Horizontal servo pin
    vertical.attach(13); // Vertical servo pin
    horizontal.write(servohori);
    vertical.write(servovert);
    delay(2000);
}

void loop() {
    // ===== Read and Filter LDR Values =====
    int lt = readAverage(ldrlt); // Top Left
    int rt = readAverage(ldrrt); // Top Right
    int ld = readAverage(ldrld); // Bottom Left
    int rd = readAverage(ldrrd); // Bottom Right
    // ===== Control Parameters =====
    int tol = 40; // tolerance for movement (higher = less sensitive)
    int step = 2; // servo movement step
    int avgLight = (lt + rt + ld + rd) / 4;
    // ===== Ignore at Low Light (Night Mode) =====
    if (avgLight < 200) {
        Serial.println("Low light detected  Tracker on standby...");
        delay(500);
        return;
    }
    // ===== Compute Averages =====
```




```
int avt = (lt + rt) / 2; // average of top sensors
int avd = (ld + rd) / 2; // average of bottom sensors
int avl = (lt + ld) / 2; // average of left sensors
int avr = (rt + rd) / 2; // average of right sensors
// ===== Calculate Differences =====
int dvert = avt - avd; // vertical difference
int dhoriz = avl - avr; // horizontal difference
// ===== Vertical Adjustment =====
if (abs(dvert) > tol) {
  if (avt > avd) servover = servover + step;
  else servover = servover - step;
  servover = constrain(servover, servoverLimitLow, servoverLimitHigh);
  vertical.write(servover);
}
// ===== Horizontal Adjustment =====
if (abs(dhoriz) > tol) {
  if (avl > avr) servohor = servohor - step;
  else servohor = servohor + step;
  servohor = constrain(servohor, servohorLimitLow, servohorLimitHigh);
  horizontal.write(servohor);
}
// ===== Debug Output =====
Serial.print("TL: "); Serial.print(lt);
Serial.print(" TR: "); Serial.print(rt);
Serial.print(" BL: "); Serial.print(ld);
Serial.print(" BR: "); Serial.print(rd);
Serial.print(" | dV: "); Serial.print(dvert);
Serial.print(" dH: "); Serial.print(dhoriz);
Serial.print(" | H: "); Serial.print(servohor);
Serial.print("° V: "); Serial.println(servover);
// ===== Small Delay for Stability =====
delay(2);
}
```

Примеры вопросов для подготовки к промежуточной аттестации

Примерные вопросы для подготовки к зачету

1. Основные понятия, классификация встраиваемых систем
2. Аналоговый и цифровой сигналы



3. Электронное устройство
4. Искажение сигналов
5. Преимущества и недостатки цифровых сигналов
6. Виды цифровых сигналов
7. Типы цифровых устройств
8. Три модели цифровых устройств
9. Три типа выходов
10. Соединение выходов разных типов
11. Типы организации связей
12. Цифровые элементы, узлы, микросхемы
13. Понятия двоичной логики
14. Обзор современных встраиваемых систем
15. Основные проблемы разработки встраиваемых систем
16. Особенности и ограничения программных средств встраиваемых систем
17. Командный интерпретатор, права доступа и процессы
18. Общие принципы и примеры составления компьютерных программ для встроенных систем
19. Обзор инструментальных средств для программирования встроенных систем
20. Среда разработчика, библиотеки, трансляторы, редакторы связей, отладка
21. Средства программирования встраиваемых систем
22. Обзор средств разработки ПО для встраиваемых систем
23. Инструментальные среды разработки для микроконтроллеров семейства AVR
24. Элементы языка программирования C и C++
25. Примеры программ для AVR и Arduino
26. Основы программирования на языке Си
27. Архитектура RISC
28. Структура программы для микроконтроллера
29. Порты ввода вывода
30. Прерывания
31. Таймеры
32. Использование широтноимпульсной модуляции
33. Работа с памятью EEPROM, FLASH
34. Организация связи с персональным компьютером (интерфейс RS-232)
35. Эмуляторы и логические анализаторы
36. Эффективные шаблоны доступа к различным типам памяти микроконтроллера



37. Подсистема аналого-цифрового преобразования микроконтроллера
38. Устройства дискретного ввода: кнопки, переключатели, клавиатуры
39. Устройства индикации: светодиоды, семисегментные индикаторы, индикаторы логического выхода с тремя состояниями
40. Изучение интерфейса UART
41. Изучение интерфейса I2C и работа с цифровыми датчиками
42. Изучение интерфейса SPI и работа с цифровыми модулями
43. Операционные системы реального времени для встраиваемых систем
44. Особенности встраиваемых систем на базе Linux
45. Отличия «встраиваемых» Linux-систем и систем на базе Android от Desktop-версий
46. Построение ядра встраиваемой операционной системы
47. Объяснение основ программирования под Arduino
48. Средства программирования Arduino
49. Обучение языку программирования
50. Режимы обмена информацией с периферийными устройствами (ПУ)
51. Программный ввод-вывод, ввод-вывод по прерываниям, прямой доступ к оперативной памяти
52. Изучение основных типов данных
53. Изучение основных функций
54. Изучение остальных основных компонент синтаксиса языка
55. Рассмотрение синтаксиса языка на конкретных примерах
56. Работа в режимах прерывания и захвата
57. Рассмотрение некоторых методов принятия решения
58. Инструментальные программные и аппаратные модули National Instruments
59. Аппаратные платформы National Instruments: NI CompactRIO, программируемый промышленный контроллер на базе ПЛИС
60. NI roboRIO, роботизированный контроллер, используемый в стандарте FIRST Robotics Competition NI CompactDAQ
61. Системы сбора данных для USB и Ethernet
62. Модельное проектирование встраиваемых систем в LabVIEW



8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Кудрявцев, Н. Г. Элементарные основы программирования встраиваемых систем [Электронный ресурс] : учебное пособие / Кудрявцев Н. Г. Горно-Алтайск : ГАГУ, 2021 148 с. Книга из коллекции ГАГУ - Информатика <https://e.lanbook.com/book/178005>. [БИК ID: RU-LAN-BOOK-178005]

Дополнительная литература:

1. Крутских, Владислав Викторович Моделирование в LabVIEW : учебник для вузов / В. В. Крутских. Электрон. дан. Москва : Юрайт, 2026 171 с (Высшее образование) URL: <https://urait.ru/bcode/588426> (дата обращения: 27.02.2026). Режим доступа: Электронно-библиотечная система Юрайт, для авториз. пользователей <https://urait.ru/bcode/588426> ISBN 978-5-534-13681-4 : 819.00. [БИК ID: RU2fURAIT2f588426]
2. Рогожников, Е. В. Программирование встраиваемых систем [Электронный ресурс] : методические указания для выполнения лабораторных работ для студентов направления подготовки 11.03.02 по дисциплине программирование встраиваемых систем / Рогожников Е. В., Савенко К. В., Гмырь В. Москва : ТУСУР, 2021 102 с. Книга из коллекции ТУСУР - Информатика <https://e.lanbook.com/book/313193>. [БИК ID: RU-LAN-BOOK-313193]

Нормативные правовые акты:

-

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Электронная библиотека Финансового университета (ЭБ) <http://elib.fa.ru/> (<http://library.fa.ru/files/elibfa.pdf>)
2. Электронно-библиотечная система BOOK.RU <http://www.book.ru>
3. Электронно-библиотечная система "Университетская библиотека ОНЛАЙН" <http://biblioclub.ru/>
4. Электронно-библиотечная система Znanium <http://www.znaniy.com>
5. • Электронно-библиотечная система издательства «ЮРАЙТ» <https://urait.ru/>
6. • Электронно-библиотечная система издательства Проспект <http://ebs.prospekt.org/books>



7. Электронно-библиотечная система издательства "Лань" <https://e.lanbook.com/>
8. • Деловая онлайн-библиотека Alpina Digital <http://lib.alpinadigital.ru/>
9. Электронная библиотека Издательского дома «Гребенников» <https://grebennikon.ru/>
10. Математические журналы: полнотекстовая коллекция Математического института им. В.А. Стеклова РАН <https://www.mathnet.ru/>
11. Научная электронная библиотека eLibrary.ru <http://elibrary.ru>
12. • Национальная электронная библиотека <http://нэб.рф/>
13. Ресурсы информационно-аналитического агентства по финансовым рынкам Cbonds.ru <https://cbonds.ru/>
14. СПАРК <https://spark-interfax.ru/>
15. • CNKI. Academic Reference <https://ar.oversea.cnki.net/>
16. • Электронные продукты издательства Elsevier <http://www.sciencedirect.com>
17. Emerald: Management eJournal Portfolio <https://www.emerald.com/insight/>
18. Реферативная база данных по математике MathSciNET <https://mathscinet.ams.org/mathscinet/>
19. • Коллекция научных журналов Oxford University Press <https://academic.oup.com/journals/>
20. • Электронные коллекции книг и журналов издательства Springer: <http://link.springer.com/>
21. Платформа STATISTA <https://www.statista.com/>
22. • База данных научных журналов издательства Wiley <https://onlinelibrary.wiley.com/>



10. Методические указания для обучающихся по освоению дисциплины

Основные этапы работы студента по дисциплине **«Программирование для встраиваемых систем»**:

1. Предварительная ориентировка в подлежащем изучению учебном материале по программе.
2. Ознакомление с рекомендованной учебной литературой.
3. Слушание и конспектирование лекций, а также выполнение других видов учебной работы.
4. Планирование самостоятельной работы.
5. Обобщение и систематизация информации, почерпнутой из лекций и прочитанной литературы.
6. Выполнение контрольной работы.

При подготовке контрольной работы следует использовать нормативные документы Финансового университета, Методические рекомендации по планированию и организации внеаудиторной самостоятельной работы студентов по образовательным программам бакалавриата и магистратуры в Финансовом университете, утвержденные приказом Финуниверситета от 11.05.2021 г. № 1040/о (см. сайт Финансового Университета: на главной странице раздел "Наш университет" → "Единая правовая база Финуниверситета" → "Организация учебного процесса" → "Нормативные документы по самостоятельной работе").

Рекомендации по работе с учебным материалом:

1. Осознавайте наличный уровень полученных вами знаний.
2. В ситуации непонимания нужно выявить тот первичный уровень и факторы непонимания, которые стали препятствием понимания последующего.
3. Задавайте сами себе вопросы и пытайтесь ответить на них.

Рекомендации по работе на лекции и с лекционным материалом:

1. Основная задача на лекции – осмысление излагаемого в ней материала. Для этого необходимо слушать лекцию с самого начала, не упуская общих, ориентирующих в материале рассуждений и установок лектора.
2. Ведение записей на лекции важно и полезно для лучшего осмысливания материала, для сохранения информации, с целью ее дальнейшего использования.
3. Для облегчения записи рекомендуется применять сокращения повторяющихся терминов или хорошо известных понятий.



Рекомендации по работе с литературой:

1. Если возникли затруднения при разыскивании материала, по какому- либо конкретному вопросу, следует обратиться к предметному указателю, напечатанному, как правило, в конце каждого литературного источника.

2. Предметный указатель это алфавитный список основных научных понятий (терминов), содержание которых раскрыто в книге, рядом с термином стоят числа, обозначающие номера страниц, на которых изложен материал, относящийся к данному понятию.



11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

Комплект лицензионного программного обеспечения:

1. Операционная система (Windows / macOS / Linux)
2. Visual Studio Code IDE для разработки, редактирование Markdown, работа с Git
3. Microsoft Windows 10/11 Professional Операционная система рабочих станций
4. Антивирус Kaspersky

Современные профессиональные базы данных и информационные справочные системы

1. Scopus
2. Информационно-правовая система «Гарант»
3. Информационно-правовая система «Консультант Плюс»
4. Электронная энциклопедия: <http://ru.wikipedia.org/wiki/Wiki>
5. Система комплексного раскрытия информации «СКРИН» - <http://www.skrin.ru/>

Сертифицированные программные и аппаратные средства защиты информации

1. не предусмотрены

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

1. **Компьютерный класс** для проведения учебных занятий, предусмотренных программой, в том числе групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенный оборудованием и техническими средствами обучения: мебель аудиторная (столы, стулья, доска аудиторная), персональные компьютеры, набор демонстрационного оборудования (проектор, экран)
2. **Учебная аудитория** для проведения учебных занятий, предусмотренных программой, в том числе групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная оборудованием и техническими средствами обучения: мебель аудиторная (столы, стулья, доска аудиторная), набор демонстрационного оборудования (проектор, экран)