

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Пекаревский Борис Владимирович
Должность: Проректор по учебной и методической работе
Дата подписания: 09.11.2023 16:34:07
Уникальный программный ключ:
3b89716a1076b80b2c167df0f27c09d01782ba84



МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Санкт-Петербургский государственный технологический институт
(технический университет)»

УТВЕРЖДАЮ
Проректор по учебной
и методической работе
_____ Б.В. Пекаревский
«22» мая 2023 г.

Рабочая программа дисциплины
ПРОГРАММИРОВАНИЕ НА ЯЗЫКАХ НИЗКОГО УРОВНЯ

Направление подготовки
09.03.01 Информатика и вычислительная техника

Направленность программы бакалавриата
Автоматизированные системы обработки информации и управления

Квалификация

Бакалавр

Форма обучения

Заочная

Факультет **информационных технологий и управления**

Кафедра **системного анализа и информационных технологий**

Санкт-Петербург

2023

Б1.В.01

ЛИСТ СОГЛАСОВАНИЯ

| Должность разработчика | Подпись | Ученое звание, фамилия, инициалы |
|------------------------|---------|-------------------------------------|
| Доцент | | А.В. Гайков |

Рабочая программа дисциплины «Программирование на языках низкого уровня» обсуждена на заседании кафедры системного анализа и информационных технологий протокол от «07» марта 2023 года № 4
Заведующий кафедрой профессор А.А. Мусаев

Одобрено учебно-методической комиссией факультета информационных технологий и управления протокол от «17» мая 2023 года № 7
Председатель доцент В.В. Куркина

СОГЛАСОВАНО

| | | |
|--|--|------------------------|
| Руководитель направления подготовки «Информатика и вычислительная техника» | | доцент И.В. Новожилова |
| Директор библиотеки | | Т.Н. Старостенко |
| Начальник методического отдела учебно-методического управления | | М.В. Труханович |
| Начальник учебно-методического управления | | С.Н. Денисенко |

СОДЕРЖАНИЕ

| | |
|--|----|
| 1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы..... | 4 |
| 2. Место дисциплины в структуре образовательной программы..... | 5 |
| 3. Объем дисциплины | 5 |
| 4. Содержание дисциплины | 6 |
| 4.1. Разделы дисциплины и виды занятий | 6 |
| 4.2. Занятия лекционного типа..... | 6 |
| 4.3. Занятия семинарского типа | 8 |
| 4.3.1. Семинары, практические занятия | 8 |
| 4.4. Самостоятельная работа обучающихся | 9 |
| 4.5. Темы контрольных работ..... | 10 |
| 5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине | 10 |
| 6. Фонд оценочных средств для проведения промежуточной аттестации | 10 |
| 7. Перечень учебных изданий, необходимых для освоения дисциплины | 11 |
| 8. Перечень электронных образовательных ресурсов, необходимых для освоения дисциплины..... | 11 |
| 9. Методические указания для обучающихся по освоению дисциплины | 11 |
| 10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине | 12 |
| 10.1. Информационные технологии | 12 |
| 10.2. Программное обеспечение | 12 |
| 10.3. Базы данных и информационно справочные системы | 12 |
| 11. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине | 12 |
| 12. Особенности освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья..... | 12 |

Приложения: 1. Фонд оценочных средств для проведения промежуточной аттестации.

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения образовательной программы обучающийся должен овладеть следующими результатами обучения по дисциплине:

| Код и наименование компетенции | Код и наименование индикатора достижения компетенции | Планируемые результаты обучения (дескрипторы) |
|--|--|--|
| ПК-5 Способен разрабатывать требования и проектировать программное обеспечение | ПК-5.13 Разработка и адаптация программного обеспечения на языках низкого уровня | Знать: - основные требования, синтаксис и принципы разработки и адаптации программного обеспечения на языках низкого уровня (ЗН-1). Уметь: - разрабатывать консольные приложения и приложения с графическим пользовательским интерфейсом на языке низкого уровня (У-1). Владеть: - навыками разработки приложений на языке низкого уровня (Н-1). |

2. Место дисциплины в структуре образовательной программы

Дисциплина относится к дисциплинам части, формируемой участниками образовательных отношений (Б1.В.01) и изучается на установочной и зимней сессиях на 1 курсе.

Изучение данной дисциплины базируется на знании студентами основ математики, информатики и основ алгоритмизации в пределах программы средней школы.

Полученные в процессе изучения дисциплины «Программирование на языках низкого уровня» знания, умения и навыки могут быть использованы при изучении дисциплин «Программирование на языке Python», «Разработка программного обеспечения», а также в научно-исследовательской работе бакалавра и при выполнении выпускной квалификационной работы.

3. Объем дисциплины

| Вид учебной работы | Всего, академических часов | |
|--|-------------------------------|----------------------|
| | 1 курс установочная сессия | 1 курс зимняя сессия |
| Общая трудоемкость дисциплины (зачетных единиц/ академических часов) | 1/36 | 2/72 |
| Контактная работа с преподавателем: | 4 | 6 |
| занятия лекционного типа | 4 | - |
| занятия семинарского типа, в т.ч. | - | 6 |
| семинары, практические занятия (в т.ч. практическая подготовка) | - | 6 |
| лабораторные работы | - | - |
| курсовое проектирование (КР или КП) | - | - |
| КСР | - | - |
| другие виды контактной работы | - | - |
| Самостоятельная работа | 32 | 62 |
| Форма текущего контроля (Кр, реферат, РГР, эссе) | - | Кр (2 шт) |
| Форма промежуточной аттестации (КР, КП, зачет, экзамен) | - | Зачет (4) |

4. Содержание дисциплины

4.1. Разделы дисциплины и виды занятий

| № п/п | Наименование раздела дисциплины | Занятия лекционного типа, академ. часы | Занятия семинарского типа, академ. часы | | Самостоятельная работа, академ. часы | Формируемые компетенции | Формируемые индикаторы |
|-------|---|--|---|---------------------|--------------------------------------|-------------------------|------------------------|
| | | | Семинары и/или практические занятия | Лабораторные работы | | | |
| 1. | Архитектура процессора | 0.25 | 0.5 | | 8 | ПК-5 | ПК-5.13 |
| 2 | Введение в язык Ассемблера | 0.25 | | | 8 | ПК-5 | ПК-5.13 |
| 3. | Структура программы на языке Ассемблера | 0.5 | 1 | | 8 | ПК-5 | ПК-5.13 |
| 4. | Арифметические и логические операции | 0.5 | 1 | | 8 | ПК-5 | ПК-5.13 |
| 5. | Управление памятью ЭВМ | 0.5 | 0.5 | | 12 | ПК-5 | ПК-5.13 |
| 6. | Основные управляющие конструкции | 0.5 | 1 | | 16 | ПК-5 | ПК-5.13 |
| 7. | Файловая система. Операции с файлами | 0.5 | 0.5 | | 12 | ПК-5 | ПК-5.13 |
| 8. | Интерфейс API | 0.5 | 1 | | 14 | ПК-5 | ПК-5.13 |
| 9 | Дизассемблирование и отладка | 0.5 | 0.5 | | 8 | ПК-5 | ПК-5.13 |
| | Итого: | 4 | 6 | | 94 | | |

4.2. Занятия лекционного типа

| № раздела дисциплины | Наименование темы и краткое содержание занятия | Объем, академ. часы | Инновационная форма |
|----------------------|--|---------------------|---------------------|
| 1 | <u>Архитектура процессора</u> Арифметические регистры. Регистр флагов. Указатель команд. Сегментные регистры. Режимы адресации. Системы счисления. Десятичная, двоичная, восьмеричная и шестнадцатеричная системы счисления. Команды микропроцессора. | 0.25 | ЛВ |

| № раздела дисциплины | Наименование темы и краткое содержание занятия | Объем, акад. часы | Инновационная форма |
|----------------------|---|-------------------|---------------------|
| 2 | <u>Введение в язык Ассемблер</u> Особенности языка Ассемблер. Команды. Псевдооператоры. Метки. Процесс компиляции, компоновки, отладки. Назначение языка ассемблер в современных условиях. | 0.25 | ЛВ |
| 3 | <u>Введение в язык Ассемблер</u> Особенности языка Ассемблер. Команды. Псевдооператоры. Метки. Процесс компиляции, компоновки, отладки. Назначение языка ассемблер в современных условиях. | 0.5 | ЛВ |
| 4 | <u>Арифметические и логические операции</u> Базовые операции. Побитовые операции. Операции с переносом. Оператор сравнения. | 0.5 | ЛВ |
| 5 | <u>Управление памятью ЭВМ</u> Структура памяти. Модели памяти. Работа с динамической памятью. | 0.5 | ЛВ |
| 6 | <u>Основные управляющие конструкции</u> Команды безусловного перехода. Команды условного перехода. Циклические процессы. Вызовы подпрограмм. Прерывания. | 0.5 | ЛВ |
| 7 | <u>Файловая система. Операции с файлами</u> Взаимодействие дисковой системы и файловой системы. Режимы открытия файлов. Файлы последовательного и произвольного доступа. Дескриптор. | 0.5 | ЛВ |
| 8 | <u>Интерфейс API</u> Программирование в среде Windows. Вызовы функций. Соглашения о передачи параметров. Наборы, функция win32 Api. Разработка dll-библиотек. | 0.5 | ЛВ |
| 9 | <u>Дизассемблирование и отладка.</u> Использование отладчика. Использование дизассемблера. Защита программ от исследования кода. | 0.5 | ЛВ |

4.3. Занятия семинарского типа

4.3.1. Семинары, практические занятия

| № раздела дисциплины | Наименование темы и краткое содержание занятия | Объем, акад. часы | Инновационная форма |
|----------------------|---|-------------------|---------------------|
| 1 | <u>Архитектура процессора</u> Работа с регистрами микропроцессора. Перевод чисел. между 10-ой, 2-й, 8-й и 16-й системами счисления. Машинные команды обмена данными и выполнения элементарных арифметических операций. | 0.5 | КтСм |
| 3 | <u>Структура программы на языке Ассемблер</u> Разработка консольных приложений, выполняющих элементарные арифметические действия. | 1 | КтСм |
| 4 | <u>Арифметические и логические операции</u> Базовые операции. Побитовые операции. Операции с переносом. Оператор сравнения. | 1 | КтСм |
| 5 | <u>Основные управляющие конструкции</u> Программная реализация разветвляющихся и циклических алгоритмов с помощью консольных приложений. | 0.5 | КтСм |
| 6 | <u>Управление памятью ЭВМ</u> Работа с динамической памятью. Массивы и структуры. Очереди, стеки, деки, связные списки и деревья. Прерывания и вызов процедур. | 1 | КтСм |
| 7 | <u>Файловая система. Операции с файлами</u> Взаимодействие дисковой системы и файловой системы. Режимы открытия файлов. Файлы последовательного и произвольного доступа. Дескриптор. | 0.5 | КтСм |
| 8 | <u>Интерфейс API</u> Программирование в среде Windows. Вызовы функций. Соглашения о передачи параметров. Наборы, функция WIN32 API. Разработка DLL-библиотек. Разработка драйверов устройств. | 1 | КтСм |
| 9 | <u>Дизассемблирование и отладка.</u> Работа с отладчиком. Использование дизассемблера. Установка и снятие защиты с программного кода. | 0.5 | КтСм |

4.4. Самостоятельная работа обучающихся

| № раздела дисциплины | Перечень вопросов для самостоятельного изучения | Объем, акад. часы | Форма контроля |
|----------------------|--|-------------------|-----------------------|
| 1 | <u>Архитектура процессора</u> Арифметические регистры. Регистр флагов. Указатель команд. Сегментные регистры. Режимы адресации. Системы счисления. Десятичная, двоичная, восьмеричная и шестнадцатеричная системы счисления. Команды микропроцессора. | 8 | Контрольная работа №1 |
| 2 | <u>Введение в язык Ассемблера</u> Особенности языка Ассемблера. Команды. Псевдооператоры. Метки. Процесс компиляции, компоновки, отладки. Назначение языка ассемблер в современных условиях. | 8 | Контрольная работа №1 |
| 3 | <u>Структура программы на языке Ассемблера</u> Модели памяти. Сегменты кода, данных, стека. Динамическая память. | 8 | Контрольная работа №1 |
| 4 | <u>Арифметические и логические операции</u> Базовые операции. Побитовые операции. Операции с переносом. Оператор сравнения. | 8 | Контрольная работа №1 |
| 5 | <u>Управление памятью ЭВМ</u> Структура памяти. Модели памяти. Работа с динамической памятью. | 12 | Контрольная работа №1 |
| 6 | <u>Основные управляющие конструкции</u> Команды безусловного перехода. Команды условного перехода. Циклические процессы. Вызовы подпрограмм. Прерывания. | 16 | Контрольная работа №1 |
| 7 | <u>Файловая система. Операции с файлами</u> Взаимодействие дисковой системы и файловой системы. Режимы открытия файлов. Файлы последовательного и произвольного доступа. Декриптор. | 12 | Контрольная работа №2 |
| 8 | <u>Интерфейс API</u> Программирование в среде Windows. Вызовы функций. Соглашения о передачи параметров. Наборы, функция win32 Api. Разработка dll-библиотек. | 14 | Контрольная работа №2 |
| 9 | <u>Дизассемблирование и отладка.</u> Работа с отладчиком. Использование дизассемблера. Установка и снятие защиты с программного кода. | 8 | Контрольная работа №2 |

4.5 Темы контрольных работ

В качестве примера содержания контрольной работы №1 по разделам 1-6, могут быть рекомендованы следующие задачи:

1. В среде Visual Studio создать пустой проект с файлом *.cpp. Используя в языке C++ директиву `_asm { }` вставить в программу кусок ассемблерного кода, реализующего вычисление выражения: $y = \frac{5 * x + 4}{7 * y - 25}$, где $x = 5$, $y = 8$.

2. В среде Visual Studio создать пустой проект с файлом *.cpp. Используя в языке C++ директиву `_asm { }` вставить в программу кусок ассемблерного кода, реализующего вычисление значения y :

$$y = \begin{cases} 9 * x - 2 * b, & \text{если } x < 7 \\ 8 * x + 5, & \text{если, } x > 7 \\ 5 * x - 7, & \text{если } x = 7 \end{cases} .$$

В качестве примера содержания контрольной работы №2 по разделам 7-9, могут быть рекомендованы следующие задачи:

1. Составить программу на языке Ассемблера, выводящую окно в операционной системе Windows. Окно должно содержать кнопку системного меню, кнопки разворачивания, свертывания и закрытия окна, а также кнопку закрытия окна.

2. Составить программу на языке Ассемблера, выводящую окно в операционной системе Windows, содержащее кнопку и текстовое поле. При вводе символов в текстовое поле и нажатии на кнопку, должно появляться диалоговое окно, выводящее содержимое текстового поля. Также должны работать кнопки: системного меню, закрытия, свертывания и разворачивания окна.

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

Методические указания для обучающихся по организации самостоятельной работы по дисциплине, включая перечень тем самостоятельной работы, формы текущего контроля по дисциплине и требования к их выполнению размещены в электронной информационно-образовательной среде СПбГТИ(ТУ) на сайте: <http://media.technolog.edu.ru>

6. Фонд оценочных средств для проведения промежуточной аттестации

Промежуточная аттестация по дисциплине проводится в виде зачета.

Зачет предусматривает выборочную проверку освоения предусмотренных элементов компетенций и комплектуется вопросами (заданиями) двух видов: теоретические вопросы (для проверки знаний) и комплексная задача (для проверки умений и навыков).

При сдаче зачета студент получает три вопроса из перечня вопросов (один из которых – задача), время подготовки студента к ответу - до 45 мин.

Пример варианта вопросов на зачете:

Вариант № 1

1. Что такое стек? Опишите принцип его работы.
2. Исследование кода программ.
3. Задача

Фонд оценочных средств по дисциплине представлен в Приложении № 1.

Результаты освоения дисциплины считаются достигнутыми, если для всех элементов компетенций достигнут пороговый уровень освоения компетенции на данном этапе – оценка «зачёт».

7. Перечень учебных изданий, необходимых для освоения дисциплины

а) печатные издания:

1. Бройдо, В. Л. Вычислительные системы, сети и телекоммуникации : Учебное пособие для вузов / В. Л. Бройдо, О. П. Ильина. – 4-е изд. – Москва ; Санкт-Петербург ; Нижний Новгород : Питер, 2011. – 554 с. – ISBN 978-5-49807-875-5.

2. Шевченко, В. П. Вычислительные системы, сети и телекоммуникации : учебник для вузов / В. П. Шевченко ; Министерство образования и науки Российской Федерации, Московский авиационный институт (национальный исследовательский университет). – Москва : КноРус, 2012. – 288 с. : ил. - Библиогр.: с. 287-288. - ISBN 978-5-406-00521-7.

б) электронные учебные издания:

3. Максимов, А. В. Оптимальное проектирование ассемблерных программ математических алгоритмов: лабораторный практикум : Учебное пособие для вузов / А. В. Максимов, Е. А. Максимова. – Санкт-Петербург [и др.] : Лань, 2022. – 128 с. – ISBN 978-5-8114-2545-7 : // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com> (дата обращения: 06.03.2023). – Режим доступа: по подписке.

4. Максимов, А. В. Оптимальное проектирование ассемблерных программ математических алгоритмов: теория, инженерные методы : Учебное пособие для вузов / А. В. Максимов. – 3-е изд., стер. – Санкт-Петербург [и др.] : Лань, 2021. – 192 с. - ISBN 978-5-8114-8056-2 : // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com> (дата обращения: 06.03.2023). – Режим доступа: по подписке.

8. Перечень электронных образовательных ресурсов, необходимых для освоения дисциплины

учебный план, РПД и учебно-методические материалы: <http://media.technolog.edu.ru>

электронно-библиотечные системы:

«Электронный читальный зал – БиблиоТех» <https://technolog.bibliotech.ru/>;

«Лань» <https://e.lanbook.com/books/>.

9. Методические указания для обучающихся по освоению дисциплины

Все виды занятий по дисциплине «Программирование на языках низкого уровня» проводятся в соответствии с требованиями следующих СТП:

СТП СПбГТИ 040-02. КС УКДВ. Виды учебных занятий. Лекция. Общие требования;

СТО СПбГТИ 018-2014. КС УКДВ. Виды учебных занятий. Семинары и практические занятия. Общие требования к организации и проведению.

СТП СПбГТИ 048-2009. КС УКДВ. Виды учебных занятий. Самостоятельная планируемая работа студентов. Общие требования к организации и проведению.

Планирование времени, необходимого на изучение данной дисциплины, лучше всего осуществлять на весь семестр, предусматривая при этом регулярное повторение пройденного материала.

Основными условиями правильной организации учебного процесса для студентов является:
плановость в организации учебной работы;
серьезное отношение к изучению материала;
постоянный самоконтроль.

На занятия студент должен приходить, имея знания по уже изученному материалу.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

10.1. Информационные технологии

В учебном процессе по данной дисциплине предусмотрено использование информационных технологий: чтение лекций с использованием слайд-презентаций; взаимодействие с обучающимися посредством ЭИОС.

10.2. Программное обеспечение

Программы Microsoft Office (Microsoft Word, Microsoft PowerPoint), интегрированная среда Microsoft Visual Studio, пакет MASM, операционная система MS Windows.

10.3. Базы данных и информационно справочные системы

Справочно-поисковая система «Консультант-Плюс»

11. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Для ведения лекционных и практических занятий используется компьютерный класс, оснащенный объединенными в сеть персональными компьютерами, оборудованием и техническими средствами обучения на необходимое количество посадочных мест.

12. Особенности освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья

Для инвалидов и лиц с ограниченными возможностями учебный процесс осуществляется в соответствии с Положением об организации учебного процесса для обучения инвалидов и лиц с ограниченными возможностями здоровья СПбГТИ(ТУ), утвержденным ректором 28.08.2014г.

**Фонд оценочных средств
для проведения промежуточной аттестации
по дисциплине «Программирование на языках низкого уровня»**

1. Перечень компетенций и этапов их формирования.

| Индекс компетенции | Содержание | Этап формирования |
|--------------------|--|-------------------|
| ПК-5 | Способен разрабатывать и адаптировать прикладное программное обеспечение | начальный |

2. Показатели и критерии оценивания компетенций на различных этапах их формирования, шкала оценивания

| Код и наименование индикатора достижения компетенции | Показатели сформированности (дескрипторы) | Критерий оценивания | УРОВНИ СФОРМИРОВАННОСТИ (описание выраженности дескрипторов) | | |
|---|---|-----------------------------------|--|--|--|
| | | | «удовлетворительно» (пороговый) | «хорошо» (средний) | «отлично» (высокий) |
| ПК-5.13 Разработка и адаптация программного обеспечения на языках низкого уровня | Рассказывает основные требования, синтаксис и принципы разработки программного обеспечения на языках низкого уровня (ЗН-1). | Ответы на вопросы №1- 51 к зачету | Путается в синтаксисе языка Ассемблера и принципах разработки программного обеспечения на языке низкого уровня | Рассказывает синтаксис языка Ассемблера и принципы разработки программного обеспечения на языке низкого уровня с небольшими ошибками | Уверенно и без ошибок рассказывает синтаксис языка Ассемблера и принципы разработки программного обеспечения на языке низкого уровня. |
| | Объясняет принципы создания консольных приложений и приложений с графическим пользовательским интерфейсом на языке низкого уровня (У-1). | | С ошибками объясняет принципы создания консольных приложений на языке Ассемблера | С небольшими погрешностями объясняет принципы создания консольных приложений и приложений с графическим пользовательским интерфейсом на языке Ассемблера | Объясняет принципы создания консольных приложений и приложений с графическим пользовательским интерфейсом на языке Ассемблера без ошибок, приводит примеры |
| | Демонстрирует навыки разработки приложений на языке низкого уровня (Н-1). | | Имеет слабые навыки разработки приложений на языке низкого уровня | Демонстрирует навыки разработки приложений на языке низкого уровня, но допускает 1-2 ошибки | Демонстрирует уверенные навыки разработки приложений на языке низкого уровня |

Шкала оценивания соответствует СТО СПбГТИ (ТУ). По дисциплине промежуточная аттестация проводится в форме зачета, шкала оценивания — «зачтено», «незачтено».

3. Типовые контрольные задания для проведения промежуточной аттестации

а) Вопросы для оценки знаний, умений и навыков, сформированных у студента по компетенции ПК-5:

1. Напишите битовые представления для следующих чисел: 5, 13, 21, 27, 75
2. Определите двоичные дополнения для следующих двоичных чисел: 00010011, 00111100, 00111001.
3. Переведите число 157 из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную.
4. Переведите число 00111100 из двоичной системы счисления в десятичную, восьмеричную и шестнадцатеричную.
5. Какие регистры можно использовать для сложения и вычитания?
6. Какие регистры можно использовать для подсчета числа циклов?
7. Какие регистры можно использовать для умножения и деления?
8. Какие регистры можно использовать для адресации сегментов?
9. Какие регистры можно использовать для индикации нулевого результата?
10. Какие регистры можно использовать для адресации выполняемой команды?
11. Перечислите основные особенности языка Ассемблер.
12. Команды и псевдооператоры.
13. Компиляция, компоновка и отладка программы. Какие программы для этого используются.
14. Где используется язык ассемблера в современных условиях?
15. Структура программы на ассемблере.
16. Какие модели памяти Вам известны?
17. Что такое сегмент кода. Для чего он используется?
18. Что такое сегмент данных. Для чего он используется?
19. Что такое сегмент стека. Для чего он используется?
20. Основные арифметические операции
21. Побитовые операции сравнения.
22. Логические операции. Оператор сравнения
23. Напишите программу на ассемблере для сложения двух чисел
24. Напишите программу на ассемблере для сложения вычитания двух чисел
25. Напишите программу на ассемблере для умножения двух чисел
26. Напишите программу на ассемблере для деления двух чисел
27. Структура памяти ПК. Модели памяти
28. Дальние и ближние вызовы
29. Что такое стек? Опишите принцип его работы.
30. Что такое очередь? Опишите принцип ее работы
31. Как реализуются команды безусловного перехода?
32. Как реализуются команды условного перехода?
33. Как осуществляется программирование циклов?
34. Подпрограммы. Вызов подпрограмм. Работа со стеком.
35. Прерывания. Виды прерываний
36. Файловая система FAT.
37. Файловая система NTFS.
38. Взаимодействие дисковой системы и файловой системы
39. Режимы открытия файлов.
40. Дескриптор.
41. Файлы последовательного доступа.
42. Файлы произвольного доступа.
43. Для чего используется API?
44. Как вызываются функции API?

45. Как осуществляется передача параметров при работе с функциями API?
46. Написание программ с графическим интерфейсом Windows.
47. Принципы написания драйверов.
48. Принципы разработки DLL-библиотек.
49. Отладчик. Выбор и использование.
50. Дизассемблирование.
51. Исследование кода программ.

При сдаче зачета, студент получает два вопроса из перечня, приведенного выше и одну задачу.

Пример задачи: Составить программу для определения суммы трех положительных чисел.

Время подготовки студента к устному ответу на вопросы - до 45 мин.

5. Методические материалы для определения процедур оценивания знаний, умений и навыков, характеризующих этапы формирования компетенций.

Промежуточная аттестация по дисциплине проводится в соответствии с требованиями СПбГТИ(ТУ) 016-2015. КС УКДВ Порядок проведения зачетов и экзаменов.