

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
ФГБОУ ВО «Уральский государственный экономический университет»

Одобрено
на заседании педагогического совета
колледжа

23 апреля 2020 г.
протокол № 9

Директор колледжа



А.Э. Чечулин



Утверждено
советом по учебно-методическим вопросам
и качеству образования

20 мая 2020 г.
протокол № 9



Д.А. Карх

КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА ДИСЦИПЛИНЫ

Наименование дисциплины	Основы алгоритмизации и программирования
Наименование специальности	09.02.07 Информационные системы и программирование
Форма обучения	Очно-заочная
Год набора	2020

Разработано
преподавателем

В.В. Городничевым

Екатеринбург
2020

1. ОБЩАЯ ХАРАКТЕРИСТИКА

Контрольно-оценочные средства (КОС) предназначены для контроля и оценки образовательных достижений обучающихся.

КОС включают контрольные материалы для проведения текущего контроля и промежуточной аттестации.

Результатом освоения дисциплины является формирование у обучающихся следующих компетенций:

ОК 1	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
ОК 2	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности
ОК 4	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами
ОК 5	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста
ОК 9	Использовать информационные технологии в профессиональной деятельности
ОК 10	Пользоваться профессиональной документацией на государственном и иностранном языке
ПК 1.1	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
ПК 1.2	Разрабатывать программные модули в соответствии с техническим заданием
ПК 1.3	Выполнять отладку программных модулей с использованием специализированных программных средств
ПК 1.4	Выполнять тестирование программных модулей
ПК 1.5	Осуществлять рефакторинг и оптимизацию программного кода
ПК 2.4	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения
ПК 2.5	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования

В результате освоения дисциплины обучающийся должен иметь:

Умения	Знания
<ul style="list-style-type: none"> - разрабатывать алгоритмы для конкретных задач; - использовать программы для графического отображения алгоритмов; - определять сложность работы алгоритмов; - работать в среде программирования; - реализовывать построенные алгоритмы в виде программ на конкретном языке программирования; - оформлять код программы в соответствии со стандартом кодирования; - выполнять проверку, отладку кода про- 	<ul style="list-style-type: none"> - понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции; - эволюцию языков программирования, их классификацию, понятие системы программирования; - основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; - подпрограммы, составление библиотек подпрограмм

граммы	- объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения
--------	---

2. МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ

Список тем рефератов

- 1 Формализация алгоритма.
- 2 Представление алгоритма в виде блок-схемы.
- 3 Составление спецификации программы.
- 4 Использование побитовых операций.
- 5 Определение приоритетов операций и порядка их вычислений в выражениях.
- 6 Составление простейших программ.
- 7 Составление программ ввода/вывода данных.
- 8 Форматированный ввод-вывод на консоль.
- 9 Составление программ линейной структуры.
- 10 Составление программ разветвляющейся структуры.
- 11 Составление программ разветвляющейся усложненной структуры.
- 12 Составление программ циклической структуры.
- 13 Составление программ усложненной структуры.
- 14 Обработка одномерных массивов.
- 15 Обработка двумерных массивов.
- 16 Использование стандартных функций для работы с массивами.
- 17 Разработка программ с использованием указателей.
- 18 Размещение массивов в динамической памяти.
- 19 Работа со строковыми переменными.
- 20 Использование стандартных функций для работы со строками.
- 21 Работа с данными типа множество.
- 22 Разработка программ со структурированными типами данных.
- 23 Разработка усложненных программ со структурированными типами данных.
- 24 Разработка программ с использованием объединений.
- 25 Разработка программ с использованием перечисляемого типа.

3. МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Форма промежуточной аттестации: 1-й семестр - **зачет**; 2-й семестр - **экзамен**.

Вопросы к зачету

- 1 Алгоритм. Понятие алгоритма. Свойства алгоритмов. Общие принципы построения алгоритмов. Понятие алгоритмизации.
- 2 Основные алгоритмические конструкции. Базовые структуры алгоритмов: линейные, разветвляющиеся, циклические. Правила построения блок-схем алгоритмов.
- 3 Эволюция и классификация языков программирования. Сущность методов программирования: структурного, модульного, объектно-ориентированного. Понятие ин-

тегрированной системы программирования. Основные принципы объектноориентированного программирования, понятие классов и объектов, их свойств и методов.

4 Жизненный цикл программного обеспечения. Основные этапы решения задач на ЭВМ. Постановка задачи и спецификация программы. Общие принципы разработки программного обеспечения. Анализ программы. Корректность программы.

5 Алфавит и лексика языка программирования. Элементы языка программирования. Управляющие последовательности. Ключевые слова. Идентификатор. Структура программы.

6 Стандартные типы данных. Понятие типа данных. Категории типов данных. Основные типы данных. Спецификаторы типов. Преобразования типов в операторе присваивания и в выражениях. Приведение типов.

7 Константы. Переменные и классы памяти. Выражения и операции. Типы констант. Правила определения констант. Определение переменной. Инициализация переменных. Спецификаторы класса памяти.

8 Операции арифметические, сравнения, логические и побитовые. Операции присваивания, последовательного вычисления, составные операторы присваивания, условная операция. Приоритеты операций и порядок вычислений.

9 Система ввода/вывода языка программирования. Функции для ввода/вывода символов, строк. Форматированный ввод/вывод на консоль. Спецификаторы и модификаторы формата.

10 Операторы выражения, составного, безусловного перехода. Синтаксис операторов: выражения, составного, безусловного перехода.

11 Операторы ветвления. Синтаксис операторов полного и неполного ветвления. Оператор множественного выбора.

12 Операторы циклов. Синтаксис операторов циклов: с параметром, предусловием и постусловием.

13 Массивы. Понятие одномерного и многомерного массивов. Синтаксис объявления массива. Правила ввода и вывода одномерных и двумерных массивов.

14 Массивы символов. Особенности ввода и вывода символьных массивов. Операции над числовыми и символьными массивами. Стандартные функции для работы с массивами.

15 Указатели. Синтаксис объявления ссылок и указателей. Способы обращения к элементам массивов с помощью указателей и ссылок. Стандартные функции для размещения переменных в динамической памяти.

16 Множество. Синтаксис объявления пользовательского типа данных множество. Особенности использования переменных типа множество.

17 Перечисление. Синтаксис объявления пользовательского типа данных перечисление. Особенности вывода значений переменных типа перечисление. Оператор sizeof, назначение оператора. Возможности оператора typedef.

18 Структуры. Синтаксис объявления структур. Доступ к элементу структуры. Массивы структур.

19 Объединения, битовые поля. Синтаксис объявления объединений, битовых полей. Доступ к элементу объединения, битовому полю.

20 Подпрограммы, правила составления подпрограмм и библиотек программ.

Вопросы к экзамену

1. Языки программирования, их классификация.
2. Системы программирования.
3. Основные конструкции алгоритмического языка

4. Жизненный цикл программы
5. Структура программы в языке
6. Опишите структуру окна среды программирования QBasic?
7. Опишите структуру окна среды программирования PascalABC?
8. Правила записи арифметических выражений.
9. Как сохранить программу на диске?
10. Как открыть в QBasic и PascalABC файл с программой?
11. Буквы какого алфавита используются в языке QBasic и PascalABC ?
12. Какие типы данных вам известны?
13. Что такое переменная? Что такое константа?
14. Что может быть именем переменной?
15. Как обозначается константа?
16. Какие типы переменных существуют? Как обозначаются переменные разных типов?
17. О чем говорит тип переменной?
18. Для чего служит оператор присваивания?
19. Как оформляется оператор присваивания? Как он работает (что происходит при его выполнении)?
20. Какова синтаксическая структура оператора ввода языка QBasic?
21. Назначение и формы записи оператора INPUT ?
22. Назначение и формы записи операторов DATA, READ, RESTORE?
23. Назначение и форматы оператора PRINT?
24. Как оформляется оператор ввода?
25. Что можно указывать в качестве элементов списка ввода?
26. Как работает оператор ввода (что происходит при его выполнении)?
27. Как оформляется оператор вывода на экран?
28. Что можно указывать в качестве элементов списка вывода?
29. Какой символ используется для разделения элементов списка вывода?
30. С помощью какого оператора языка QBasic и PascalABC осуществляется вывод значений переменных, текста, а также значений арифметических выражений?
31. Что будет выведено на экран, если в списке вывода записано:
 - a) число?
 - b) имя величины?
 - c) текст в кавычках?
 - d) арифметическое выражение?
32. Каково предназначение оператора, состоящего из одного ключевого слова END?
33. Какие виды условных операторов вы знаете?
34. В каких случаях в программе используется полный условный оператор?
35. Как он оформляется? Как он работает (что происходит при его выполнении)?
36. В каких случаях в программе используется неполный условный оператор? Как он работает (что происходит при его выполнении)?
37. Дать определение разветвляющегося алгоритма.
38. Привести примеры разветвляющихся алгоритмов.
39. Назовите виды разветвляющихся алгоритмов.
40. С помощью какого оператора осуществляется запись разветвляющегося алгоритма в программе на языке QBasic и PascalABC?
41. В чем состоит отличие в синтаксической структуре условного блочного оператора и условного строчного оператора?

42. В каких разветвляющихся алгоритмах существует возможность использования условного строчного оператора?
43. Какой блок обязательно присутствует при записи разветвляющегося алгоритма в виде блок-схемы?
44. Какими видами выражений языка QBasic и PascalABC записывается условие в условных операторах?
45. Дать определение циклического алгоритма.
46. Привести примеры циклических алгоритмов.
47. Назовите виды циклических алгоритмов.
48. С помощью каких операторов осуществляется запись циклического алгоритма в программе на языке QBasic?
49. В чем состоит отличие в выполнении цикла с предусловием от выполнения цикла с постусловием?
50. Какова синтаксическая структура оператора цикла с параметром?
51. Какой блок обязательно присутствует в блок-схеме при записи циклического алгоритма с конечным количеством повторений?
52. Допустимо ли изменение параметра цикла в теле цикла при использовании оператора цикла с параметром?
53. Какой оператор используется для записи вложенных циклов? В чем состоит синтаксическая особенность его использования?
54. Принципы модульного программирования
55. Описание функции
56. Прототипы функций.
57. Что такое «массив», «элемент массива», «индекс элемента массива»?
58. Почему массив является структурированным типом данных?
59. Какого типа могут быть элементы массива?
60. Какого типа могут быть индексы элементов массива?
61. Какие простые типы данных относятся к порядковым?
62. Что такое «размер» и «размерность» массива? Назовите виды массивов
63. Объявление и инициализация одномерного массива.
64. Объявление и инициализация двумерного массива.
65. Доступ к элементам одномерного массива по индексу.
66. Доступ к элементам одномерного массива по адресу.
67. Доступ к элементам двумерного массива по индексу.
68. Доступ к элементам двумерного массива по адресу.
69. Чем отличается запись числовых переменных от записи символьных переменных?
70. Что такое длина слова?
71. Какая функция используется для определения длины слова?
72. Какая функция используется для выделения из слова его части (слева, справа, середины)?
73. Дать определение понятию файловой переменной.
74. Стандартные процедуры и функции обработки данных файла.
75. Формат файловой переменной для текстовых файлов. Перечислить процедуры и функции для работы с текстовыми файлами.
76. Формат файловой переменной для нетипизированных файлов. Перечислить процедуры и функции для работы с файлами без типа.
77. Назначение и формат операторов SCREEN, VIEW, WINDOW.
78. Назначение и формат операторов LINE, CIRCLE.
79. Назначение и формат операторов PSET, PRESET.

80. Назначение и формат оператора DRAW.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Критерии оценки (недифференцированной)	Критерии оценки (дифференцированной)
<p>Оценка «зачтено» выставляется студенту, проявившему знания основного программного материала в объеме, необходимом для последующего обучения и предстоящей практической деятельности, знакомому с основной рекомендованной литературой, но допустившему неполные или слабо аргументированные ответы, испытывающему затруднения.</p> <p>Оценка «не зачтено» выставляется студенту, обнаружившему значительные пробелы в знании программного материала, допустившему принципиальные ошибки при применении теоретических знаний, которые не позволяют ему продолжить обучение или приступить к практической деятельности без дополнительной подготовки по данной дисциплине</p>	<p>«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.</p> <p>«Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.</p> <p>«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.</p> <p>«Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки</p>