

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Тобольский педагогический институт им. Д.И. Менделеева (филиал)
Тюменского государственного университета

УТВЕРЖДАЮ
Директор

Шилов С.П.



ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по дисциплине ПОО 01. Основы алгоритмизации
для обучающихся по программе подготовки специалистов среднего звена
09.02.05 Прикладная информатика (по отраслям)
(базовая подготовка)
Форма обучения – очная

Зыбина Н.В. ПОО 01. Основы алгоритмизации. Фонд оценочных средств дисциплины для обучающихся по программе подготовки специалистов среднего звена 09.02.05 Прикладная информатика (по отраслям). Форма обучения – очная. Тобольск, 2020.

Фонд оценочных средств дисциплины разработаны на основе ФГОС СПО по специальности 09.02.05 Прикладная информатика (по отраслям), утвержденного приказом Министерства образования и науки Российской Федерации от 13 августа 2014 года, № 1001.

© Тобольский педагогический институт им. Д.И. Менделеева (филиал) Тюменского государственного университета, 2020

© Зыбина Н.В, 2020

Содержание

1. Общая характеристика фонда оценочных средств	3
2. Паспорт фонда оценочных средств	5
3. Типовые задания для оценки освоения учебной дисциплины	9

1. ОБЩАЯ ХАРАКТЕРИСТИКА ФОНДОВ ОЦЕНОЧНЫХ СРЕДСТВ

1.1. Область применения программы

Фонд оценочных средств учебной дисциплины «Основы алгоритмизации» является частью программы подготовки специалистов среднего звена в соответствии с ФГОС СПО по специальности 09.02.05 Прикладная информатика (по отраслям).

Фонд оценочных средств предназначен для проверки результатов освоения учебной дисциплины «Основы алгоритмизации» и может быть использован в профессиональной подготовке студентов по квалификации – техник-программист.

1.2. Место дисциплины в структуре программы подготовки специалистов среднего звена

Дисциплина «Основы алгоритмизации», входит в блок общеобразовательной подготовки, предлагаемые ОО.

1.3. Цели и задачи дисциплины – требования к результатам освоения дисциплины.

Освоение содержания дисциплины обеспечивает достижение студентами следующих **результатов:**

личностных:

- чувство гордости и уважения к истории развития и достижениям отечественной информатики в мировой индустрии информационных технологий;
- осознание своего места в информационном обществе;
- готовность и способность к самостоятельной и ответственной творческой деятельности с использованием информационно-коммуникационных технологий;
- умение использовать достижения современной информатики для повышения собственного интеллектуального развития в выбранной профессиональной деятельности, самостоятельно формировать новые для себя знания в профессиональной области, используя для этого доступные источники информации;
- умение выстраивать конструктивные взаимоотношения в командной работе по решению общих задач, в том числе с использованием современных средств сетевых коммуникаций;
- умение управлять своей познавательной деятельностью, проводить самооценку уровня собственного интеллектуального развития, в том числе с использованием современных электронных образовательных ресурсов;
- умение выбирать грамотное поведение при использовании разнообразных средств информационно-коммуникационных технологий как в профессиональной деятельности, так и в быту;
- готовность к продолжению образования и повышению квалификации в избранной профессиональной деятельности на основе развития личных информационно-коммуникационных компетенций;

метапредметных:

- умение определять цели, составлять планы деятельности и определять средства, необходимые для их реализации;
- использование различных видов познавательной деятельности для решения информационных задач, применение основных методов познания (наблюдения, описания, измерения, эксперимента) для организации учебно-исследовательской и проектной деятельности с использованием информационно-коммуникационных технологий;
- использование различных информационных объектов, с которыми возникает необходимость сталкиваться в профессиональной сфере в изучении явлений и процессов;
- использование различных источников информации, в том числе электронных библиотек, умение критически оценивать и интерпретировать информацию, получаемую из различных источников, в том числе из сети Интернет;
- умение анализировать и представлять информацию, данную в электронных форматах на компьютере в различных видах;

- умение использовать средства информационно-коммуникационных технологий в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности;
- умение публично представлять результаты собственного исследования, вести дискуссии, доступно и гармонично сочетая содержание и формы представляемой информации средствами информационных и коммуникационных технологий;

предметных:

- сформированность представлений о роли информации и информационных процессов в окружающем мире;
- владение навыками алгоритмического мышления и понимание методов формального описания алгоритмов, владение знанием основных алгоритмических конструкций, умение анализировать алгоритмы;
- использование готовых прикладных компьютерных программ по профилю подготовки;
- владение способами представления, хранения и обработки данных на компьютере;
- владение компьютерными средствами представления и анализа данных в электронных таблицах;
- сформированность представлений о базах данных и простейших средствах управления ими;
- сформированность представлений о компьютерно-математических моделях и необходимости анализа соответствия модели и моделируемого объекта (процесса);
- владение типовыми приемами написания программы на алгоритмическом языке для решения стандартной задачи с использованием основных конструкций языка программирования;
- сформированность базовых навыков и умений по соблюдению требований техники безопасности, гигиены и ресурсосбережения при работе со средствами информатизации;
- понимание основ правовых аспектов использования компьютерных программ и прав доступа к глобальным информационным сервисам;
- применение на практике средств защиты информации от вредоносных программ, соблюдение правил личной безопасности и этики в работе с информацией и средствами коммуникаций в Интернете.

Умения	Знания
У1. Определять тип вычислительного процесса. У2. Записывать алгоритмы. У3. Разрабатывать алгоритмы для решения задач, работы программы (блок-схемы). У4. Решать задачи тестирования и отладки программного обеспечения. У5. Использовать инструментальные среды реализации алгоритмов. У6. Идентифицировать, анализировать и структурировать данные. У7. Владеть навыками алгоритмического мышления и понимание методов формального описания алгоритмов. У8. Умение анализировать алгоритмы.	31. Свойства алгоритма: конечность, определенность, результативность, массовость. 32. Область определения алгоритма. 33. Знание основных алгоритмических конструкций алгоритмов: линейные, разветвляющиеся, циклические. 34. Базовые и динамические структуры данных и операции над ними. 35. Вспомогательные алгоритмы. 36. Классификацию языков программирования и их функциональное назначение.

2. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

п/п	Темы дисциплины, МДК, разделы (этапы) практики, в ходе текущего контроля, вид промежуточной аттестации с указанием семестра	Код контролируемой компетенции (или её части), знаний, умений	Наименование оценочного средства (с указанием количества вариантов, заданий и т.п.)
1.	Тема 1.1. Основы понятия алгоритмизации	31, 32, У2	Тест (8 вопросов)
2.	Тема 1.2. Основные алгоритмические конструкции.	31, 32, 33, У1, У2, У3	Контрольная работа (3 варианта)
3.	Тема 1.3. Данные: понятие и типы	31, 32, 33, 34, У1, У2, У3, У4	Сообщение (темы 10)
4.	Тема 1.4. Вспомогательные алгоритмы.	31, 32, 33, 34, 35, У1, У2, У3, У4	Сообщения-презентация (6 тем)
5.	Тема 2.1. Основные принципы программирования	34, 35, 36, У1, У5, У6	Тест (10 вопросов)
6.	Промежуточная аттестация в 2 семестре	31, 32, 33, 34, 35, 36, У1, У2, У3, У4, У5, У6	Дифференцированный зачет

3. ТИПОВЫЕ ЗАДАНИЯ ДЛЯ ОЦЕНКИ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Тема 1.1. Основы понятия алгоритмизации	31, 32, У2
---	------------

Тест «Основы понятия алгоритмизации»

1. По линии прямой связи передаются

- 1) команды управления и информация об объекте управления
- 2) информация о состоянии объекта управления
- 3) информация о состоянии управляющей системы
- 4) команды управления
- 5) команды управления и информация об управляющей системе

2. Какой из объектов может являться исполнителем алгоритмов?

- 1) карта
- 2) словарь
- 3) стиральная машина
- 4) тетрадь
- 5) краска

3. Понятность алгоритма означает, что он должен быть записан с помощью:

- 1) команд, понятных создателю алгоритма
- 2) команд из системы команд исполнителя
- 3) команд, понятых пользователю алгоритма
- 4) команд, понятных для компьютера
- 5) операторов языка программирования

4. Конечность алгоритма означает, что:

- 1) в нем должен присутствовать оператор вывода результата
- 2) он должен решать задачу вычислительного характера
- 3) в нем должно присутствовать ключевое слово, означающее конец алгоритма
- 4) он должен быть применим для решения всех задач заданного типа
- 5) результат должен быть получен за конечное число шагов

5. Как называется свойство алгоритма, соответствующее определению: «Алгоритм должен быть записан из команд, понятных исполнителю, каждая команда должна определять однозначное действие исполнителя»?

- 1) массовость
- 2) точность
- 3) конечность
- 4) понятность
- 5) дискретность

6. Суть такого свойства алгоритма как результативность заключается в том, что:

- 1) алгоритм должен иметь дискретную структуру (должен быть разбит на последовательность отдельных шагов);
- 2) записывая алгоритм для конкретного исполнителя, можно использовать лишь те команды, что входят в систему его команд;
- 3) алгоритм должен обеспечивать решение не одной конкретной задачи, а некоторого класса задач данного типа;
- 4) при точном исполнении всех команд алгоритма процесс должен прекратиться за конечное число шагов, приведя к определенному результату;
- 5) исполнитель алгоритма не должен принимать решения, не предусмотренные составителем алгоритма.

7. Свойство алгоритма как понятность заключается в том, что:

- 1) алгоритм должен иметь дискретную структуру (должен быть разбит на последовательность отдельных шагов);
- 2) записывая алгоритм для конкретного исполнителя, можно использовать лишь те команды, что входят в систему его команд;

- 3) алгоритм должен обеспечивать решение не одной конкретной задачи, а некоторого класса задач данного типа;
 - 4) исполнитель алгоритма не должен принимать решения, не предусмотренные составителем алгоритма.
- 8. Суть такого свойства алгоритма как массовость заключается в том, что:**
- 1) алгоритм должен иметь дискретную структуру (должен быть разбит на последовательность отдельных шагов);
 - 2) записывая алгоритм для конкретного исполнителя, можно использовать лишь те команды, что входят в систему его команд;
 - 3) алгоритм должен обеспечивать решение не одной конкретной задачи, а некоторого класса задач данного типа;
 - 4) при точном исполнении всех команд алгоритма процесс должен прекратиться за конечное число шагов, приведя к определенному результату;
 - 5) исполнитель алгоритма не должен принимать решения, не предусмотренные составителем алгоритма.

Тема 1.2. Основные алгоритмические конструкции.	31, 32, 33, У1, У2, У3,
---	-------------------------

Контрольная работа

Вариант № 1

1. Составить алгоритм вычисления функции: $y = \frac{2 \cdot x^2 + 4}{4 - x}$
2. Составить алгоритм табулирования функции: $y = x^3 + 2 \cdot x - 8$ на отрезке $[-a; b]$ с шагом h .
3. Составить алгоритм вычисления суммы всех членов последовательности $a_n = \frac{n+1}{2^n}$
4. Составить алгоритм нахождения корней квадратного уравнения.
5. Составить алгоритм подсчета количества четных чисел в введенном пользователем четырехзначного числа.

Вариант № 2

1. Составить программу вычисления функции: $y = \frac{2 \cdot x^3 - 12}{2 \cdot x - 6}$
2. Составить программу табулирования функции: $y = 2 \cdot x^2 - 2$ на отрезке $[-2,4; 4,2]$ с шагом 0,3.
3. Составить программу вычисления суммы всех членов последовательности $a_n = \frac{2 \cdot n}{n!}$
4. Составить программу нахождения наибольшего числа из трех введенных.
5. Составить программу подсчета суммы цифр трехзначного числа.

Вариант № 3

1. Составить программу вычисления функции: $y = \begin{cases} x^2, & x \leq 0 \\ \frac{1}{x}, & x > 0 \end{cases}$
2. Составить программу табулирования функции: $y = 3 \cdot x^2 - 4 \cdot x$ на отрезке $[0; 12]$.
3. Составить программу вычисления суммы всех членов последовательности $a_n = \frac{2 \cdot n}{2 \cdot n!}$
4. Написать программу нахождения суммы большего и меньшего из трех чисел.
5. Дан круг радиуса R . Определить, поместится ли правильный треугольник со стороной a в этом круге.

Тема 1.3. Данные: понятие и типы	31, 32, 33, 34, У1, У2, У3, У4
----------------------------------	--------------------------------

Темы сообщений

1. Обход в глубину и в ширину
2. Алгоритм Дейкстры
3. Алгоритм Флойда
4. Алгоритм Форда-Белмана
5. Ориентированные графы
6. Каркасы и деревья
7. Алгоритмы нахождения кратчайшего пути
8. Трехмерные массивы
9. Зубчатые массивы
10. Динамические структуры данных

Тема 1.4. Вспомогательные алгоритмы.	31, 32, 33, 34, 35, У1, У2, У3, У4
--------------------------------------	------------------------------------

Темы сообщений-презентаций

1. Примеры рекурсивных алгоритмов
2. Реализация механизма рекурсии
3. Сравнение рекурсии и итерации
4. Рекурсивные алгоритмы с заглядыванием вперед
5. Рекурсивные алгоритмы с возвратом
6. Рекурсия вокруг нас: литература, физика, математика, головоломки

Тема 2.1. Основные принципы программирования	34, 35, 36, У1, У5, У6
--	------------------------

Тест:

1. Языки программирования условно разделены на классы: машинные, машинно-ориентированные, алгоритмические, процедурно-ориентированные, проблемно-ориентированные. По какому признаку классифицированы языки?
 - 1) синтаксис образования конструкций языков программирования
 - 2) принадлежность к одному из оформившихся к настоящему времени стилей программирования
 - 3) уровень языка программирования, характеризующий степень его близости к машинному языку
2. Языки программирования условно разделены на классы: процедурные, функциональные, логические, объектно-ориентированные. По какому признаку классифицированы языки?
 - 1) синтаксис образования конструкций языков программирования
 - 2) принадлежность к одному из оформившихся к настоящему времени стилей программирования
 - 3) уровень языка программирования, характеризующий степень его близости к машинному языку
3. К языкам высокого уровня не относятся
 - 1) Ada
 - 2) язык Ассемблера
 - 3) двоичный язык
 - 4) C/C++

4. Примером функциональных языков программирования может служить
 - 1) LISP
 - 2) PROLOG
 - 3) DELPHI
 - 4) PASCAL
5. Примером процедурных языков программирования может служить
 - 1) LISP
 - 2) PROLOG
 - 3) DELPHI
 - 4) PASCAL
6. Примером логических языков программирования может служить
 - 1) LISP
 - 2) PROLOG
 - 3) DELPHI
 - 4) PASCAL
7. Программа, на основе которой машина преобразует вводимые в нее команды на машинный язык, называется
 - 1) переводчиком
 - 2) транслятором
 - 3) системным администратором
 - 4) редактором связей
8. Транслятор, который переводит каждую команду программы с одновременным её выполнением и, если обнаруживает ошибку, сообщает о ней и прекращает выполнение программы, называется
 - 1) компилятором
 - 2) редактором связей
 - 3) системным администратором
 - 4) интерпретатором
9. Транслятор, который переводит всю программу целиком и в конце работы выдаёт список ошибок, если они обнаружены, называется
 - 1) компилятором
 - 2) редактором связей
 - 3) системным администратором
 - 4) интерпретатором
10. К языкам высокого уровня относятся
 - 1) Ada
 - 2) язык Ассемблера
 - 3) двоичный язык
 - 4) C/C++

Промежуточная аттестация во 2 семестре	31, 32, 33, 34, 35, 36, У1, У2, У3, У4, У5, У6
--	---

Вопросы к дифференцированному зачету

1. Понятие алгоритма.
2. Свойства алгоритмов.

3. Исполнители алгоритма
4. Формы записей алгоритмов.
5. Алгоритмы линейной структуры
6. Алгоритмы ветвления
7. Алгоритмы циклической структуры
8. Алгоритмы на графах
9. Алгоритмы обхода графа
10. Алгоритмы нахождения кратчайшего пути
11. Данные. Основные базовые типы данных и их характеристика.
12. Структурированные типы данных и их характеристика.
13. Одномерные массивы. Перебор элементов массива
14. Двумерные массивы. Сортировка элементов массива.
15. Трехмерные массивы
16. Зубчатые массивы
17. Динамические структуры данных
18. Вспомогательные алгоритмы. Подпрограммы.
19. Рекурсивные подпрограммы.
20. Решение логических задач
21. Алгоритмически неразрешимые проблемы
22. Развитие языков программирования. Классификация языков программирования.
23. Обзор языков программирования. Области применения языков программирования. Стандарты языков программирования. Среда проектирования. Компиляторы и интерпретаторы.
24. Основы структурного программирования. Развитие языков и технологий программирования. Структура и способы описания языков программирования высокого уровня
25. Основные этапы решения задач на компьютере.