

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
Учебно-методическое объединение по педагогическому образованию

УТВЕРЖДАЮ

Первый заместитель Министра
образования Республики Беларусь
_____ А.Г.Баханович

Регистрационный № _____

МЕТОДЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЕ

**Примерная учебная программа по учебной дисциплине
для специальности
6-05-0113-04 Физико-математическое образование
(физика и информатика; математика и информатика; информатика)**

СОГЛАСОВАНО

Председатель учебно-методического
объединения по педагогическому
образованию

_____ А.И.Жук

СОГЛАСОВАНО

Начальник Главного управления
профессионального образования
Министерства образования
Республики Беларусь

_____ С.Н.Пищов

СОГЛАСОВАНО

Начальник Главного управления
общего среднего, дошкольного
и специального образования
Министерства образования
Республики Беларусь

_____ М.С.Киндиренко

СОГЛАСОВАНО

Проректор по научно-методической
работе Государственного учреждения
образования «Республиканский
институт высшей школы»

_____ И.В.Титович

Эксперт-нормоконтролер

СОСТАВИТЕЛИ:

А.А.Францкевич, заведующий кафедрой информатики и методики преподавания информатики физико-математического факультета учреждения образования «Белорусский государственный педагогический университет имени Максима Танка», кандидат педагогических наук;

С.И.Зенько, доцент кафедры информатики и методики преподавания информатики физико-математического факультета учреждения образования «Белорусский государственный педагогический университет имени Максима Танка», кандидат педагогических наук, доцент;

С.В.Вабищевич, доцент кафедры информатики и методики преподавания информатики физико-математического факультета учреждения образования «Белорусский государственный педагогический университет имени Максима Танка», кандидат педагогических наук, доцент;

А.З.Кутыш, старший преподаватель кафедры информатики и методики преподавания информатики физико-математического факультета учреждения образования «Белорусский государственный педагогический университет имени Максима Танка»

РЕЦЕНЗЕНТЫ:

Кафедра дискретной математики и алгоритмики факультета прикладной математики и информатики Белорусского государственного университета (протокол № 20 от 15.06.2023);

В.С.Якимович, доцент кафедры информационных технологий в культуре учреждения образования «Белорусский государственный университет культуры и искусств», кандидат педагогических наук, доцент

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ В КАЧЕСТВЕ ПРИМЕРНОЙ:

Кафедрой информатики и методики преподавания информатики физико-математического факультета учреждения образования «Белорусский государственный педагогический университет имени Максима Танка» (протокол № 16 от 27.06.2023);

Научно-методическим советом учреждения образования «Белорусский государственный педагогический университет имени Максима Танка» (протокол № 7 от 11.07.2023);

Научно-методическим советом по физико-математическому образованию и технологии учебно-методического объединения по педагогическому образованию (протокол № 2 от 12.07.2023)

Ответственный за редакцию: А.А. Францкевич

Ответственный за выпуск: А.А. Францкевич

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Примерная учебная программа по учебной дисциплине «Методы алгоритмизации и программирование» разработана для учреждений высшего образования в соответствии с требованиями образовательного стандарта общего высшего образования по специальности 6-05-0113-04 «Физико-математическое образование (физика и информатика; математика и информатика; информатика)».

Содержание учебной дисциплины «Методы алгоритмизации и программирование» предполагает использование в работе со студентами современных образовательных технологий и подходов к разработке алгоритмов обработки различных структур данных и решения практико-ориентированных заданий из разных предметных областей. Научно-методические идеи взаимосвязанного обучения и деятельностно-семантического подхода при работе с учебным контентом содействуют развитию у студентов логического, алгоритмического и творческого мышления, а также способствуют подготовке их к активной профессиональной деятельности в условиях современного общества.

Целью учебной дисциплины является формирование профессиональной компетентности учителя информатики, связанной с систематизацией и обобщением имеющихся знаний и способов деятельности по разработке алгоритмов (полученных в результате обучения информатике в школе), а также усвоением современных подходов к введению и изучением основных понятий методов алгоритмизации и программирования.

Задачи учебной дисциплины:

- формирование знаний теоретических основ алгоритмизации;
- овладение понятийным аппаратом языков и технологий программирования;
- изучение основных принципов структурного программирования;
- приобретение умений разработки алгоритмов обработки данных и их реализации.

Учебная дисциплина «Методы алгоритмизации и программирование» входит в модуль «Технологии обработки информации» и тесно связана с изучением учебных дисциплин «Компьютерная графика и мультимедиа», «Современные компьютерные технологии». Результаты изучения учебной дисциплины способствуют успешному овладению содержанием учебных дисциплин «Программирование в визуализированных средах», «Системы и технологии программирования», «Практикум по решению задач по информатике».

В результате изучения учебной дисциплины студент должен:

знать:

- основные понятия алгоритмизации;
- специальную терминологию парадигмы структурного программирования;

- основные структуры данных и базовые алгоритмические конструкции;
- методы разработки алгоритмов для обработки различных структур данных;
- сущность и особенности структурного программирования;
- основные приемы использования современных технологий программирования при разработке алгоритмов для решения практико-ориентированных заданий;

уметь:

- разрабатывать алгоритмы для решения учебных заданий;
- работать в современных средах программирования;
- использовать основные конструкции языка программирования высокого уровня при проектировании и отладке алгоритмов;
- преобразовывать разработанный алгоритм в программу средствами языка программирования высокого уровня;
- разрабатывать алгоритмы решения практико-ориентированных заданий;

владеть:

- способами проектирования алгоритмов и их реализацией с помощью современных средств программирования;
- приемами разработки алгоритмов для обработки различных структур данных;
- методами и средствами программирования в соответствии с парадигмой структурного программирования.

Освоение учебной дисциплины «Методы алгоритмизации и программирование» должно обеспечить формирование **универсальной компетенции**: решать стандартные задачи профессиональной деятельности на основе применения информационно-коммуникационных технологий; **базовой профессиональной компетенции**: применять методы, способы и средства создания, обработки и хранения информации с использованием современного прикладного программного обеспечения, методы и технологии алгоритмизации и программирования для реализации учебного процесса.

В рамках образовательного процесса по данной учебной дисциплине студент должен приобрести не только теоретические и практические знания, умения и навыки по специальности, но и развить свой ценностно-личностный, духовный потенциал, сформировать качества патриота и гражданина, готового к активному участию в экономической, производственной, социально-культурной и общественной жизни страны.

Всего на изучение учебной дисциплины «Методы алгоритмизации и программирование» отводится 216 часов, из них 112 аудиторных. Примерное распределение аудиторных часов по видам занятий: 40 часов – лекции, 56 часов – лабораторные, 16 часов – практические.

Рекомендуемые формы промежуточной аттестации – зачет, экзамен.

ПРИМЕРНЫЙ ТЕМАТИЧЕСКИЙ ПЛАН

№	Наименование раздела, темы	Всего аудиторных часов	из них		
			лекции	практические	лабораторные
1.ОСНОВЫ ТЕОРИИ И ПРАКТИКИ АЛГОРИТМИЗАЦИИ И ЯЗЫКОВ ПРОГРАММИРОВАНИЯ		6	6		
1.1.	Основы алгоритмизации	4	4		
1.2.	Основы теории языков программирования	2	2		
2.МЕТОДЫ РАЗРАБОТКИ АЛГОРИТМОВ В СТРУКТУРНОМ ПРОГРАММИРОВАНИИ		106	34	16	56
2.1.	Основные элементы языка программирования. Типы переменных, выражения	6	4		2
2.2.	Операторы языка программирования. Базовые алгоритмические конструкции. Алгоритмы обработки данных скалярных (простых) типов	32	8	6	18
2.3.	Основы программирования графики	8	2		6
2.4.	Структурированные (составные, сложные) типы данных. Алгоритмы обработки данных структурированных типов	38	12	8	18
2.5.	Вспомогательные алгоритмы и подпрограммы	12	4	2	6
2.6.	Основы работы с файлами	10	4		6
Всего:		112	40	16	56

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

РАЗДЕЛ 1. ОСНОВЫ ТЕОРИИ И ПРАКТИКИ АЛГОРИТМИЗАЦИИ И ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

Тема 1.1. Основы алгоритмизации

Этапы решения задач на компьютере.

Понятие алгоритма. Свойства алгоритма. Способы представления алгоритмов.

Блок-схемы алгоритмов, основные элементы.

Тема 1.2. Основы теории языков программирования

Понятие языка программирования. Область применения языков программирования. Критерии эффективности языков программирования.

Пути эволюции языков и технологий программирования. Парадигмы программирования. Классификация языков программирования.

РАЗДЕЛ 2. МЕТОДЫ РАЗРАБОТКИ АЛГОРИТМОВ В СТРУКТУРНОМ ПРОГРАММИРОВАНИИ

Тема 2.1. Основные элементы языка программирования. Типы переменных, выражения

Основные понятия языка программирования: «символ», «лексема» (идентификаторы, знаки операций, константы, разделители), «переменная».

Скалярные (простые) типы данных: порядковые (перечисляемый, целый, логический, символьный, диапазонный) и вещественный.

Операции и выражения. Унарные и бинарные операции. Приоритетность операций. Арифметические и логические выражения. Совместимость типов операндов выражения.

Основные арифметические функции.

Структура программы.

Тема 2.2. Операторы языка программирования. Базовые алгоритмические конструкции. Алгоритмы обработки данных скалярных (простых) типов

Оператор присваивания. Комментарии в программе. Процедуры ввода и вывода. Форматы вывода числовых данных.

Алгоритмическая конструкция «Следование».

Понятие структурного оператора. Составной оператор.

Алгоритмическая конструкция «Ветвление». Оператор «Альтернатива». Полная и неполная формы оператора «Альтернатива». Оператор «Выбор». Полная и неполная формы оператора «Выбор».

Алгоритмическая конструкция «Повторение». Понятия «цикл», «тело цикла», «условие цикла», «параметр цикла». Цикл с определяемым количеством повторений. Оператор цикла *while*. Оператор цикла *repeat*. Цикл с известным количеством повторений. Оператор цикла *for*. Вложенные циклы.

Тема 2.3. Основы программирования графики

Основы работы с графикой. Модуль для работы с графикой. Графические примитивы. Работа с цветом. Работа с пером и кистью. Имитация движения графических объектов. Построение графиков элементарных функций.

Тема 2.4. Структурированные (составные, сложные) типы данных. Алгоритмы обработки данных структурированных типов

Понятия «массив», «размер массива», «размерность массива», «одномерный массив», «двумерный массив» («матрица»), «квадратная матрица», «главная диагональ квадратной матрицы», «побочная диагональ квадратной матрицы». Описание одномерного и двумерного массивов. Способы ввода и вывода элементов одномерного и двумерного массивов.

Поиск в массиве. Методы поиска элемента массива: поиск методом полного перебора, бинарный поиск (поиск методом деления пополам) и др. Поиск элементов в одномерных и двумерных массивах с заданными свойствами (свойства элементов, лежащих на (под, над) главной и побочной диагоналями квадратной матрицы и др.).

Перестановка элементов массива. Реверсирование элементов массива. Циклические перестановки элементов массива.

Сортировка элементов массива. Методы упорядочивания элементов массива: сортировка выбором, сортировка обменом (пузырьковая) и др.

Понятия «строка», «длина строки», «подстрока». Объявление строковых величин. Операции, процедуры и функции работы со строковыми величинами.

Понятия «множество», «базовый тип множества». Описание множеств. Операции над множествами.

Понятия «запись», «поле записи». Описание записей. Массивы записей. Оператор *with*.

Тема 2.5. Вспомогательные алгоритмы и подпрограммы

Понятия «вспомогательный алгоритм», «подпрограмма». Виды подпрограмм. Локальные и глобальные переменные.

Процедуры пользователя. Структура процедуры пользователя. Организация вызова процедуры пользователя. Типы параметров: фактические и формальные; параметры-значения (получаемые параметры) и параметры-переменные (возвращаемые параметры). Виды процедур пользователя: процедуры без параметров; процедуры с получаемыми, но без возвращаемых параметров; процедуры с получаемыми и возвращаемыми параметрами.

Функции пользователя. Структура функции пользователя. Организация вызова функции пользователя. Отличия функции пользователя от процедуры пользователя.

Тема 2.6. Основы работы с файлами

Понятие файлов. Виды файлов. Типизированные файлы. Работа с типизированными файлами: создание файла, использование данных из файла, дополнение файла новыми данными.

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

ОСНОВНАЯ ЛИТЕРАТУРА

1. Городко, С. И. Современные технологии программирования : учеб.-метод. пособие для студентов / С. И. Городко, С. В. Снисаренко ; М-во образования Респ. Беларусь, Белорус. гос. ун-т информатики и радиоэлектроники. – Минск : БГУИР, 2017. – 67 с.
2. Вабищевич, С. В. Рабочая тетрадь по программированию на языке Паскаль / С. В. Вабищевич; М-во образования Респ. Беларусь, Белорус. гос. пед. ун-т. – Минск : БГПУ, 2021. – 40 с.
3. Зенько, С. И. Реализация современных образовательных технологий при обучении программированию будущих учителей информатики : пособие / С. И. Зенько, А. З. Кутыш. – 2-е изд. – Минск : БГПУ, 2020. – 320 с.

ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

4. Ахо, А. В. Разработка и анализ компьютерных алгоритмов / А. В. Ахо, Д. Э. Хопкрофт, Д. Д. Ульман — Киев: «Диалектика», 2021. – 544 с.
5. Буславский, А. А. Начальный уровень обучения программированию на языке Pascal / А. А. Буславский ; Мин. обл. ин-т развития образования. – Минск : МОИРО, 2010. – 89 с.
6. Златопольский, Д. М. Программирование: типовые задачи, алгоритмы, методы / Д. М. Златопольский. – М. : Лаборатория знаний, 2020. – 226 с.
7. Информатика : учеб. пособие для 10 кл. учреждений общ. сред. образования с рус. яз. обучения / В. М. Котов [и др.]. – Минск : Нар. асвета, 2020. – 120 с.
8. Информатика : учеб. пособие для 9 кл. учреждений общ. сред. образования с белорус. и рус. яз. обучения / В. М. Котов [и др.]. – Минск : Нар. асвета, 2019. – 156 с.
9. Кормен, Т. Х. Алгоритмы: построение и анализ / Т. Х. Кормен [и др.]. – М. : Вильямс, 2013. – 1328 с.
10. Луридаc, П. Алгоритмы для начинающих: теория и практика для разработчика / П. Луридаc – М. : Эксмо, 2018. – 608 с.
11. Окулов, С. М. Программирование в алгоритмах / С. М. Окулов. – М. : БИНОМ, 2017. – 383 с.
12. Расолько, Г. А. Система тестов для самоподготовки по курсу «Методы программирования и информатика». Язык Pascal : пособие для студентов механико-математического факультета специальности 1-31 03 01-02 «Математика (научно-педагогическая деятельность)» / Г. А. Расолько, Е. В. Кремень, Ю. А. Кремень. – Минск : Белорус. гос. ун-т, 2013. – 70 с.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Содержание и формы самостоятельной работы студентов разрабатываются в соответствии с целями и задачами подготовки специалиста.

Для управления самостоятельной работой рекомендуется использовать:

- электронные средства обучения (мультимедийные презентации, электронные пособия);
- ресурсный центр физико-математического факультета;
- тестирующие программы.

При изучении учебной дисциплины рекомендуется использовать следующие формы самостоятельной работы студентов:

- подготовка к лабораторным работам;
- подготовка к практическим занятиям;
- выполнение домашних заданий в виде решения задач по отдельным темам содержания учебной дисциплины;
- подготовка практических разработок;
- написание рефератов на основе литературных источников и цифровых образовательных ресурсов;
- оформление отчетов по лабораторным работам;
- презентация результатов лабораторных работ;
- выполнение учебно-исследовательской работы.

Текущий контроль осуществляется в ходе выполнения и защиты лабораторных работ, проектов. Самостоятельная работа студента методически организуется путем выполнения домашних заданий по материалу, пройденному на практических и лабораторных занятиях.

Особое внимание необходимо обращать на организацию индивидуальной работы студента под руководством преподавателя. Эта работа должна проводиться с учетом индивидуальных особенностей каждого студента с помощью системы индивидуальных заданий, которые студент может выполнять на основе образцов, рассмотренных на лекциях.

ПЕРЕЧЕНЬ ИСПОЛЬЗУЕМЫХ СРЕДСТВ ДИАГНОСТИКИ

Основными средствами диагностики усвоения знаний, умений и овладения необходимыми навыками по учебной дисциплине являются:

- интерактивный опрос на лекционных занятиях, направлен на систематизацию знаний студентов, определение уровня готовности аудитории к восприятию нового материала, а также на формирование у преподавателя представление об усвоении студентами основополагающих понятий и фактов изучаемой учебной дисциплины;

- проверка практических заданий (репродуктивные, продуктивные, творческие задания), выполняемых на практических и лабораторных

занятиях, представляет собой диагностику систематичности подготовки студентов к занятиям и уровня усвоения ими практико-ориентированного содержания программного материала учебной дисциплины;

– групповые и индивидуальные консультации студентов, которые предназначены для диагностики уровня овладения знаниями, умениями и навыками, устранения возможных ошибок, пробелов в знаниях студентов;

– самостоятельные работы используются для определения индивидуальных особенностей, темпа продвижения студентов и усвоения ими необходимых знаний;

– компьютерное тестирование позволяет быстро провести диагностику усвоения студентами учебного материала как по отдельным темам и разделам учебной дисциплины, так и по учебной дисциплине в целом;

– зачет (экзамен) используются для осуществления итоговой диагностики усвоения учащимися содержания учебной дисциплины за учебный семестр с оценкой в соответствии с критериями оценки результатов учебной деятельности обучающихся в учреждениях высшего образования.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ВОПРОСОВ К ЗАЧЕТУ

1. Этапы решения задач на компьютере.
2. Понятие алгоритма.
3. Свойства алгоритма.
4. Способы представления алгоритмов.
5. Блок-схемы алгоритмов, основные элементы.
6. Понятие языка программирования.
7. Область применения языков программирования.
8. Критерии эффективности языков программирования.
9. Пути эволюции языков и технологий программирования.
10. Парадигмы программирования.
11. Классификация языков программирования.
12. Основные понятия языка программирования: «символ», «лексема» (идентификаторы, знаки операций, константы, разделители), «переменная».
13. Скалярные (простые) типы данных: порядковые (перечисляемый, целый, логический, символьный, диапазонный) и вещественный.
14. Операции и выражения. Унарные и бинарные операции. Приоритетность операций.
15. Оператор присваивания. Комментарии в программе.
16. Процедуры ввода и вывода.
17. Форматы вывода числовых данных.
18. Алгоритмическая конструкция «Следование». Понятие структурного оператора. Составной оператор.
19. Алгоритмическая конструкция «Ветвление». Оператор «Альтернатива».
20. Алгоритмическая конструкция «Ветвление». Оператор «Выбор».
21. Алгоритмическая конструкция «Повторение». Понятия «цикл», «тело цикла», «условие цикла», «параметр цикла».

22. Цикл с определяемым количеством повторений. Оператор цикла while.
23. Цикл с определяемым количеством повторений. Оператор цикла repeat.
24. Цикл с известным количеством повторений. Оператор цикла for.
25. Основы работы с графикой. Модуль для работы с графикой.
26. Графические примитивы. Работа с цветом.
27. Графические примитивы. Работа с пером и кистью.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ВОПРОСОВ К ЭКЗАМЕНУ

1. Понятие алгоритма. Свойства алгоритма. Способы представления алгоритмов.
2. Блок-схемы алгоритмов, основные элементы.
3. Понятие языка программирования. Область применения языков программирования. Критерии эффективности языков программирования.
4. Основные понятия языка программирования: «символ», «лексема» (идентификаторы, знаки операций, константы, разделители), «переменная».
5. Скалярные (простые) типы данных: порядковые (перечисляемый, целый, логический, символьный, диапазонный) и вещественный.
6. Операции и выражения. Унарные и бинарные операции. Приоритетность операций.
7. Процедуры ввода и вывода. Форматы вывода числовых данных.
8. Алгоритмическая конструкция «Следование». Понятие структурного оператора. Составной оператор.
9. Алгоритмическая конструкция «Ветвление». Оператор «Альтернатива».
10. Алгоритмическая конструкция «Ветвление». Оператор «Выбор».
11. Алгоритмическая конструкция «Повторение». Понятия «цикл», «тело цикла», «условие цикла», «параметр цикла».
12. Цикл с определяемым количеством повторений. Оператор цикла while.
13. Цикл с определяемым количеством повторений. Оператор цикла repeat.
14. Цикл с известным количеством повторений. Оператор цикла for.
15. Основы работы с графикой. Модуль для работы с графикой. Графические примитивы. Работа с цветом. Работа с пером и кистью.
16. Понятия «массив», «размер массива», «размерность массива». Описание одномерных и двухмерных массивов.
17. Способы ввода и вывода элементов одномерных и двухмерных массивов.
18. Квадратная матрица. Главная и побочная диагонали квадратной матрицы.
19. Поиск в массиве. Методы поиска элемента одномерного массива.
20. Поиск элементов двухмерного массива с заданными свойствами.
21. Перестановка элементов массива.

22. Реверсирование элементов массива. Циклические перестановки элементов массива.
23. Сортировка элементов массива. Сортировка выбором.
24. Сортировка элементов массива. Сортировка обменом.
25. Понятия «строка», «длина строки», «подстрока». Объявление строковых величин. Операции, процедуры и функции работы со строковыми величинами.
26. Понятия «множество», «базовый тип множества». Описание множеств. Операции над множествами.
27. Понятия «запись», «поле записи». Описание записей. Массив записей. Оператора with.
28. Понятия «вспомогательный алгоритм», «подпрограмма». Виды подпрограмм. Локальные и глобальные переменные.
29. Процедуры пользователя. Структура процедуры пользователя. Типы параметров: фактические и формальные; параметры-значения и параметры-переменные.
30. Функции пользователя. Структура функции пользователя. Отличия функции пользователя от процедуры пользователя.
31. Понятие файлов. Виды файлов. Типизированные файлы.