

**«Приднестровский государственный университет
им. Т.Г. Шевченко»**

Рыбницкий филиал

Кафедра автоматизации технологических процессов и производств

УТВЕРЖАЮ

Директор Филиала ПГУ им. Т.Г. Шевченко
в г. Рыбница, профессор

И.А.

Павлинов

“ 25 ” 2018 г.

РАБОЧАЯ ПРОГРАММА

на 2018 – 2019 учебный год

Учебной дисциплины

«Программирование и основы алгоритмизации»

Направление подготовки:

2.15.03.04. «Автоматизация технологических процессов и производств»

Профиль подготовки

15.03.04 «Автоматизация технологических процессов и производств»

квалификация (степень) выпускника

Бакалавр

Форма обучения:

Заочная

Рыбница 2018

Рабочая программа дисциплины *«Программирование и основы алгоритмизации»* /сост. Б.К. Корлюга – Рыбница: ГОУ ВО «ПГУ им. Т.Г.Шевченко», 2018 - 17 с.

**РАБОЧАЯ ПРОГРАММА ПРЕДНАЗНАЧЕНА ДЛЯ ПРЕПОДАВАНИЯ ДИСЦИПЛИНЫ
СТУДЕНТАМ ЗАОЧНОЙ ФОРМЫ ОБУЧЕНИЯ ПО НАПРАВЛЕНИЮ ПОДГОТОВКИ 2.15.03.04
«АВТОМАТИЗАЦИЯ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ И ПРОИЗВОДСТВ»**

Рабочая программа составлена в соответствии с требованиями Федерального Государственного образовательного стандарта высшего образования по направлению подготовки 15.03.04 «Автоматизация технологических процессов и производств», утвержденного приказом Министерства образования и науки Российской Федерации от 12.03.2015г. № 200.

Составитель  /Корлюга Б.К./ старший преподаватель

1 Место дисциплины в структуре основной образовательной программы (ООП).

Дисциплина относится к базовой части блока Б.1 «Дисциплины (модули)» ФГОС ВО по направлению подготовки 2.15.03.04 АВТОМАТИЗАЦИЯ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ И ПРОИЗВОДСТВ, адресована для подготовки бакалавров по профилю подготовки «Автоматизация технологических процессов и производств» и обеспечивает теоретическую и практическую подготовку студентов в области программирования и алгоритмизации.

2 Цели и задачи дисциплины.

Целью изучения дисциплины является обучение студентов современным приемам и методам составления алгоритмов и программ, реализующих решение вычислительных задач, а также общих задач обработки информации

Задачи изучения дисциплины:

- алгоритмы и программы.
- данные. Понятие типа данных.
- логические основы алгоритмизации.
- язык программирования: эволюция и классификация.
- системы программирования.

Операторы языка.

- структурированные типы данных. Динамические данные. Процедуры и функции.

Структура программы. Модули. Организация ввода вывода данных.

- работа с файлами. Файлы данных. Объектно-ориентированный подход к программированию.

3 Требования к результатам освоения дисциплины:

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО и ООП ВО по данному направлению подготовки (специальности):

Код компетенции	Формулировка компетенции
Общекультурные компетенции (ОК)	
ОК-3	Способностью к коммуникации в устной и письменной формах на русском и иностранном языках для решения задач межличностного и межкультурного взаимодействия
ОК-5	Способностью к самоорганизации и самообразованию
Общепрофессиональные компетенции (ОПК)	
ОПК-2	Способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности
ОПК-3	Способностью использовать современные информационные технологии, технику, прикладные программные средства при решении задач профессиональной деятельности
ОПК-5	Способностью участвовать в разработке технической документации, связанной с профессиональной деятельностью
Профессиональные компетенции (ПК)	
ПК-4	Способностью участвовать в постановке целей проекта (программы), его задач при заданных критериях, целевых функциях, ограничениях, разработке структуры его взаимосвязей, определении приоритетов решения задач с учетом правовых и нравственных аспектов профессиональной деятельности, в разработке проектов изделий с учетом технологических, конструкторских, эксплуатационных, эстетических, экономических и управленческих параметров, в разработке проектов модернизации действующих производств,

	создании новых, в разработке средств и систем автоматизации, контроля, диагностики, испытаний, управления процессами, жизненным циклом продукции и ее качеством в соответствии с техническими заданиями и использованием стандартных средств автоматизации расчетов и проектирования
ПК-5	Способностью участвовать в разработке (на основе действующих стандартов и другой нормативной документации) проектной и рабочей технической документации в области автоматизации технологических процессов и производств, их эксплуатационному обслуживанию, управлению жизненным циклом продукции и ее качеством; в мероприятиях по контролю соответствия разрабатываемых проектов и технической документации действующим стандартам, техническим условиям и другим нормативным документам
ПК-20	Способностью проводить эксперименты по заданным методикам с обработкой и анализом их результатов, составлять описания выполненных исследований и подготавливать данные для разработки научных обзоров и публикаций

Знать:

- синтаксис и семантику алгоритмического языка программирования, принципы и методологию построения алгоритмов программных систем;
- принципы структурного и модульного программирования с поддержкой жизненного цикла программ, а также объектно-ориентированного программирования.
- типы данных и базовые конструкции изучаемых языков программирования;
- основные приемы программирования;
- интегрированные среды изучаемых языков программирования;
- основы объектно-ориентированного программирования.

Уметь:

- проектировать простые программные алгоритмы и реализовывать их с помощью современных средств программирования.
- составлять простые блок-схемы алгоритмов;
- составлять программы на алгоритмическом языке;
- работать в интегрированной среде изучаемого языка программирования.

Владеть:

- навыками проектирования простых программных алгоритмов и реализации их на языке программирования.
- навыками проектирования простых блок-схем.

4 Структура и содержание дисциплины

4.1. Распределение трудоемкости в з.е./часах по видам аудиторной и самостоятельной работы студентов по семестрам

4.1.

Семестр	Трудоем кость, з.е./часы	Количество часов					Форма итогового контроля
		В том числе				Самост. работы	
		Аудиторных			Практич. зан.		
Всего	Лекций	Лаб. раб.					
1	2/72	6	2	4	0	66	Курсовая работа
2	2/72	8	0	4	4	64	Экзамен
Итого:	4 / 144	14	2	8	4	130	-

4.2. Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

№ раз- дела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеауд. работа (СР)
			Л	ПЗ	ЛР	
1.	Алгоритмы и программы. Данные. Понятие типа данных. Логические основы алгоритмизации. Язык программирования: эволюция и классификация. Системы программирования. Файлы данных. Объектно-ориентированный подход к программированию	23	1		2	20
2.	Переменные и константы. Типы данных. Выражения и операции. Операторы языка. Структурированные типы данных. Динамические данные. Процедуры и функции. Структура программы. Модули. Организация ввода вывода данных. Работа с файлами	49	1		2	46
<i>Итого 1 семестр:</i>		72	2		4	66
1.	Алгоритмы и программы. Данные. Понятие типа данных. Логические основы алгоритмизации. Язык программирования: эволюция и классификация. Системы программирования. Файлы данных. Объектно-ориентированный подход к программированию	14		2	2	10
2.	Переменные и константы. Типы данных. Выражения и операции. Операторы языка. Структурированные типы данных. Динамические данные. Процедуры и функции. Структура программы. Модули. Организация ввода вывода данных. Работа с файлами	58		2	2	54
<i>Итого 2 семестр:</i>		72		4	4	64
<i>Всего:</i>		144	2	4	8	130

4.3. Тематический план по видам учебной деятельности

Лекции				
№ п/п	Номер раздела дисциплины	Объем часов	Тема лекции	Учебно- наглядные пособия
1	№1	1	Программирование и алгоритмизация: синтаксис и семантика алгоритмического языка программирования; Структурное и модульное программирование; типизация и структуризация программных данных; статические и динамические данные;	Презентации, раздаточный материал
2	№2	1	Основные понятия ООП: абстракция, инкапсуляция, класс, наследование, объект, полиморфизм, прототип. Рекурсия и итерация; сортировка и поиск Методы и средства объектно-ориентированного программирования	Презентации, раздаточный материал
Итого:		2		

Практические (семинарские) занятия

№ п/п	Номер раздела дисциплины	Объем часов	Тема практического занятия	Учебно- наглядные пособия
1	№1	2	Программирование и алгоритмизация: синтаксис и семантика алгоритмического языка программирования.	Презентации, раздаточный материал
2	№2	2	Основные понятия ООП: абстракция, инкапсуляция, класс, наследование, объект, полиморфизм, прототип. Рекурсия и итерация; сортировка и поиск	Презентации, раздаточный материал
Итого:		4		

Лабораторные работы

№ п/п	Номер раздела дисциплины	Объем часов	Тема лабораторного занятия	Учебно- наглядные пособия
1	№1	1	Создание элементарных программ Их компиляция и выполнение. Основные элементы программирования; вывод: типы данных; операций; условные операторы и циклические конструкции. Описание и определение функций в программах.	Методические указания
2	№1	1	Использование в программах структур данных (массивы, файлы). Указатели и адресная арифметика в программах динамического распределения	Методические указания
3	№1	2	Использование в программах структур и объединений.	Методические указания
4	№2	2	Разработка диалогов ввода-вывода данных.	Методические указания
5	№2	2	Разработка программных модулей. Подключение и использование программных модулей в основной программе.	Методические указания
Итого:		8		

Самостоятельная работа студента

№ п/п	Разделы и темы рабочей программы самостоятельного изучения	Перечень заданий и вопросов для самостоятельного изучения	Сроки	Объем в часах
1.	Алгоритмы и программы. Данные. Понятие типа данных. Логические основы алгоритмизации. Язык программирования: эволюция и классификация. Системы программирования. Файлы данных. Объектно-ориентированный подход к программированию	Составление плана-конспекта	1,2 семестр	30
2.	Переменные и константы. Типы данных. Выражения и операции. Операторы языка. Структурированные типы данных. Динамические данные. Процедуры и функции. Структура программы. Модули. Организация ввода вывода данных. Работа с файлами	Составление плана-конспекта	1,2 семестр	100
Итого				141/3.6

5 Примерная тематика курсовых проектов (работ):

5.2. Теоретические вопросы:

№ варианта	Вопросы:
0	1. Алгоритм и его свойства. 2. Классификация стандартных типов данных в языке Турбо Паскаль.
1	1. Изобразительные средства алгоритмов: словесный, формульно-словесный, блок-схемный. 2. Стандартные функции округления и преобразования типов в языке Турбо Паскаль
2	1. Базовые канонические структуры, используемые при проектировании алгоритмов линейных, ветвящихся и циклических вычислительных процессов. 2. Особенности записи и использования оператора цикла с предусловием в языке Турбо Паскаль.
3	1. Классификация языков программирования. Общая характеристика языка Паскаль. 2. Особенности записи и использования оператора цикла с постусловием в языке Турбо Паскаль.
4	1. Арифметические, строковые и логические выражения в языке Турбо Паскаль. Приоритеты выполнения операций в выражениях. 2. Область действия имен в программах сложной структуры. Понятие глобальных и локальных переменных.
5	1. Структура программы на языке Паскаль. 2. Особенности использования формальных и фактических параметров в процедурах и функциях.
6	1. Простой и составной операторы в языке Турбо Паскаль. 2. Описание функции в языке Турбо Паскаль. Обращение к функции.
7	1. Назначение, описание и варианты использования условного оператора и оператора выбора в языке Турбо Паскаль. 2. Стандартные математические функции, процедуры и функции порядкового типа в языке Турбо Паскаль.
8	1. Виды оператора цикла в языке Турбо Паскаль. Принципы выбора типа оператора цикла. 2. Особенности использования рекурсивных процедур и функций в языке Турбо Паскаль.
9	1. Особенности записи и использования оператора цикла типа арифметической прогрессии в языке Турбо Паскаль. 2. Особенности использования формальных и фактических параметров в процедурах и функциях.

Практические задания к курсовым проектам представлены в методических рекомендациях.

6 Образовательные технологии

В процессе освоения дисциплины «Программирование и основы алгоритмизации» используются следующие образовательные технологии:

- лекции;
- тренинги;
- компьютерные занятия;
- письменные домашние работы;
- самостоятельная работа студентов, в которую включается освоение информационных технологий и интерпретации результатов;
- консультации преподавателей.

Применение каждой формы обучения предполагает применение новых ИТ – технологий.

Система последовательных, взаимосвязанных действий, обеспечивающих усвоение содержания образования, развитие способностей студентов, овладение ими средствами самообразования и самообучения; обеспечивают цель обучения, способ усвоения и характер взаимодействия преподавателя и обучающегося; направлены на приобретение знаний, формирование умений, навыков, их закрепление и контроль

Проведение аудиторных занятий (лекций и практических работ) предполагает использование аудиовизуальных электронных и компьютерных средств мультимедиа, имеющихся в арсенале Университета.

Семестр	Вид занятия (Л, ПР, ЛР)	Используемые интерактивные образовательные технологии	Количество часов
1,2	Л	Презентации, раздаточный материал	2
	ПР	разбор конкретных ситуаций с использованием интерактивных средств	4
	ЛР	Составление программ за ПК	8
Итого:			14

7 Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

Задания для контрольных работ:

Вариант I

1. Написать программу, решающую неравенство.
 $ax^2 + b \leq 0$.

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму целых положительных чисел, кратных 4 и меньших 100.

3. Разработайте программу, формирующую квадратную матрицу $D(n,n)$, где $n \leq 15$, элементы которой определяются по формуле

$$D[i,j] = \begin{cases} \sin(i+j) & \text{при } i < j; \\ 1 & \text{при } i = j; \\ \frac{i+j}{2i+3j} & \text{при } i > j, \end{cases}$$

где i – номер строки матрицы;
 j – номер столбца матрицы.

В сформированной матрице определить максимальный и минимальный элемент и поменять их местами.

Вывести исходную матрицу, соответствующие элементы и их координаты, а также преобразованную матрицу.

4. Дана строка текста длиной из 40 символов, состоящая из слов, разделённых пробелами.

Разработать программу удаления «лишних» пробелов.

Лишними считать пробелы в начале строки до первого символа, второй и более пробелы между словами и пробелы в конце строки.

Вывести исходную и преобразованную строки и количество удалённых пробелов.

Вариант II

1. Написать программу, решающую неравенство.
 $ax^2+bx+c>0$.

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму целых положительных четных чисел, меньших 100.

3. Дана матрица $A(n,m)$, где $n,m \leq 15$.

Разработайте программу, формирующую одномерный массив $V(n)$, элементами которого должно являться количество элементов каждой строки, превышающих среднее арифметическое значение матрицы в целом. Если в строке таких элементов нет, в соответствующий элемент одномерного массива заносится 0.

Вывести исходную матрицу, значение среднего арифметического элементов матрицы и сформированный массив V .

4. Разработайте программу, которая вводит строки, содержащие фамилию, имя, отчество и год рождения, а выводит – строки, содержащие фамилию, инициалы и возраст на текущий год.

Завершение ввода – при чтении пустой строки.

Вывести исходные и преобразованные строки.

Вариант III

1. Написать программу, решающую неравенство.
 $ax^2+b>0$.

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму целых положительных нечетных чисел, меньших 200.

3. Дан массив $D(n)$, где $n \leq 10$, вещественного типа.

Разработайте программу, которая вычисляет сумму трёх первых положительных элементов заданного массива. Если таких элементов нет. Программа должна выдавать соответствующее сообщение.

Выведите на печать исходный массив и искомую сумму.

4. Разработать программу, которая определяет в строке количество слов длиннее четырёх символов. Слова разделены одним пробелом.

Вывести исходную строку, полученные слова и количество полученных слов.

Вариант IV

1. Написать программу, решающую неравенство.
 $ax^2+b\leq 0$.

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму целых положительных чисел, больших 20, меньших 100 и кратных 3.

3. Дан одномерный массив вещественных чисел $A(n)$, где $n\leq 50$.

Разработайте программу, формирующую новый массив B из элементов массива A , которые превышают среднее арифметическое значение элементов массива A , стоящих на местах с чётными индексами.

Выведите среднее арифметическое значение элементов массива A , исходный и сформированный массивы.

4. Разработайте программу, которая осуществляет замену заданной строки в отсортированном по алфавиту массиве строк, и сортирует полученный массив строк в соответствии с русским алфавитом.

Вывести исходный и преобразованный массивы.

Вариант V

1. Написать программу, решающую неравенство.
 $ax^2+bx>0$.

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму первых 10 членов ряда, в котором $a_n=\ln(n!)/n^2$.

3. Дан одномерный целочисленный массив $C(n)$, где $n\leq 40$, содержащий как положительные, так и отрицательные элементы.

Разработайте программу, которая определяет номер первого отрицательного элемента, по абсолютной величине превышающего максимальный элемент этого массива.

Выведите массив C , а также номер найденного элемента, или соответствующее сообщение, если такого элемента нет.

4. Разработайте программу, меняющую в строке местами слова с указанными номерами. Запретить ввод номеров, которые превышают количество слов в строке или равны между собой.

Вывести исходную и преобразованную строки.

Вариант VI

1. Написать программу, решающую неравенство.
 $ax^2+bx+c\leq 0$.

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму первых 15 членов ряда, в котором $a_n=n^{\ln n}/(\ln n)^n$.

3. Разработайте программу, которая формирует массив $B(n)$, где $n\leq 30$, содержащий элементы целого типа в диапазоне от -20 до 130, используя датчик случайных чисел.

В сформированном массиве определите количество и среднее арифметическое значение положительных и отрицательных элементов массива.

Выведите массив B , а также все найденные в программе величины.

4. Дана строка текста длиной из 80 символов, состоящая из слов, разделённых пробелом, в конце точка.

Разработайте программу, которая определяет номера слов, в которых содержится более трёх символов «А».

Вывести исходную строку и номера слов. Если слов с таким числом букв не окажется, вывести соответствующее сообщение.

Вариант VII

1. Написать программу, решающую систему неравенств.

$$\begin{cases} a_1x + b_1 \geq 0 \\ a_2x + b_2 \leq 0 \end{cases}$$

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму первых 12 членов ряда, в котором $a_n = n! / n^{\sqrt{n}}$.

3. Дан одномерный целочисленный массив $D(n)$, где $n \leq 50$, содержащий как положительные, так и отрицательные элементы.

Разработайте программу, которая находит сумму всех чётных элементов массива, стоящих на чётных местах, то есть, имеющих четные номера.

Выведите массив D , а также все найденные в программе величины.

4. Дана строка текста длиной из 40 символов, состоящая из слов, разделённых пробелом, в конце точка.

Разработайте программу, которая удаляет из текста слово, содержащее максимальное количество букв «В».

Вывести исходную и преобразованную строки. Если в тексте нет слов с буквой «В» – вывести соответствующее сообщение

Вариант VIII

1. Написать программу, решающую систему неравенств.

$$\begin{cases} a_1x + b_1 < 0 \\ a_2x + b_2 < 0 \end{cases}$$

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму первых 7 членов ряда, в котором $a_n = e^{-\sqrt[3]{n}}$.

3. Дана матрица $A(n,m)$, где $n,m \leq 25$.

Разработайте программу, заменяющую все чётные элементы массива, на их квадраты, а нечётные удвоить.

Вывести исходную и преобразованную матрицу.

4. Дан массив символьных строк длиной 40 символов. Строки состоят из слов, разделённых пробелом, в конце точка.

Разработайте программу, которая формирует одномерный массив B , содержащий в качестве элементов количество слов каждой строки, начинающихся на гласную букву.

Если таких слов нет, в соответствующий элемент массива занести 0.

Вывести исходный и сформированный массивы.

Вариант IX

1. Написать программу, решающую систему неравенств.

$$\begin{cases} a_1x + b_1 \geq 0 \\ a_2x + b_2 \geq 0 \end{cases}$$

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму первых 9 членов ряда, в котором $a_n = n^2 e^{-\sqrt{n}}$.

3. Дана матрица $C(n,m)$, где $n,m \leq 30$.

Разработайте программу, умножающую положительные элементы на первый элемент соответствующей строки, а отрицательные – на последний.

Вывести исходную и преобразованную матрицу.

4. Дана строка, состоящая из слов, разделённых одним пробелом. Разработайте программу, которая разбивает исходную строку на максимальной длины подстроки, размер которых не превышает заданного значения n . Перенос слов считать запрещённым. Вывести исходную и полученные строки.

Вариант X

1. Написать программу, решающую систему неравенств.

$$\begin{cases} a_1x + b_1 > 0 \\ a_2x + b_2 < 0 \end{cases}$$

Программа должна правильно выполняться при любых значениях начальных данных.

2. Напишите следующую программу.

Найти сумму первых 11 членов ряда, в котором $a_n = n! / n^n$.

3. Дана матрица $E(n, m)$, где $n, m \leq 40$.

Разработайте программу, находящую номера всех максимальных элементов в каждой строке. Переместить содержимое этого максимального элемента в начало этой строки матрицы.

Вывести исходную и преобразованную матрицу.

4. Разработайте программу, которая осуществляет поиск заданной строки в отсортированном в соответствии с русским алфавитом массиве строк $n=100$.

Вывести исходный массив и количество вхождений заданной строки.

Вопросы для Экзамена

1. Определение алгоритма. Свойства алгоритма. Формы записи алгоритмов. Примеры.
2. Запись алгоритмов блок-схемами. Основные элементы блок-схем.
3. Алгоритмы с ветвлением. Пример алгоритма.
4. Алгоритм цикла с предусловием. Пример алгоритма.
5. Алгоритм цикла с постусловием. Пример алгоритма.
6. Алгоритм цикла с управляющей переменной. Пример алгоритма.
7. Основные типы данных
8. Целый и вещественный типы данных. Операции с переменными этого типа.
9. Логический тип данных. Символьный тип данных. Операции с переменными этого типа.
10. Назовите поколения языков программирования и их характеристики.
11. Дайте определение алфавита и лексики языка программирования. Приведите пример.
12. Дайте определение синтаксиса и семантики программирования. Приведите пример.
13. Из каких частей состоит исходная программа.
14. Что такое система программирования. Назовите классы систем программирования.
15. Объясните суть процессов трансляции и компиляции.
16. Что такое библиотеки подпрограмм и для чего их используют.
17. Файл. Типы файлов.
18. Общие принципы разработки ПО.
19. Частотный принцип разработки ПО и принцип модульности.
20. Принцип функциональной избирательности при разработке ПО и принцип генерируемости.
21. Принцип функциональной избыточности при разработке ПО и принцип «по умолчанию».
22. Общесистемные принципы разработки ПО.

23. Язык программирования Turbo Pascal. Структура программы.
24. Типы данных, используемые в языке Turbo Pascal.

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Основная литература:

1. Незнанов А. А. Программирование и алгоритмизация/А. А. Незнанов. – 2010.
2. Сеницын С. В. Программирование на языке высокого уровня/С. В. Сеницын, А. С. Михайлов, О. И. Хлытчиев. – 2010.
3. Канцедаль С. А. Алгоритмизация и программирование/С. А. Канцедаль. – 2010.
4. Семакин И. Г. Основы алгоритмизации и программирования/И. Г. Семакин, А. П. Шестаков. – 2008.
5. Голицына О. Л. Основы алгоритмизации и программирования/О. Л. Голицына, И. И. Попов. – 2010.
6. Колдаев В. Д. Основы алгоритмизации и программирования/В. Д. Колдаев ; под ред. Л. Г. Гагариной. – 2009.
7. Черняк А. А. Математическое программирование. Алгоритмический подход: учебное пособие для студентов экономических специальностей учреждений, обеспечивающих получение высшего образования/ А. А. Черняк, Ж. А. Черняк, Ю. М. Метельский. - Минск: Высшая школа, 2007. - 352 с.: ил.
8. Эпштейн М. С. Практикум по программированию на языке С: учебное пособие для студентов образовательных учреждений среднего профессионального образования/ М. С. Эпштейн. - М.: Академия, 2007. - 128 с.: ил.
9. Сборник задач по высшей математике для экономистов. Аналитическая геометрия, линейная алгебра, математический анализ, теория вероятностей, математическая статистика, линейное программирование: учебное пособие для студентов вузов/ В. И. Ермаков [и др.] ; под ред. В. И. Ермакова; Рос. экон. акад. им. Г. В. Плеханова. - 2-е изд., испр. - М.: ИНФРА-М, 2007. - 575 с.

8.2. Дополнительная литература

1. Павловская Т. А. Паскаль. Программирование на языке высокого уровня: практикум : учеб. пособие для студентов вузов, обучающихся по направлению "Информатика и вычислит. техника"/ Т. А. Павловская. - Москва [и др.]: Питер, 2007. - 317 с.: ил. - (Учебное пособие).
 2. Калихман Л. И. Сборник задач по линейной алгебре и программированию/ Л. И. Калихман. - Подольск: Интеграл, 2007. - 270 с.
 3. Калихман Л. И. Сборник задач по математическому программированию/ Л. И. Калихман; [учеб. пособие для экон. специальностей вузов]. - Изд. 2-е, перераб. и доп. - Подольск: Интеграл, 2006. - 270 с.
 4. Олифер В. Г. Сетевые операционные системы: [учебник для вузов]/ В. Г. Олифер, Н. А. Олифер. - 2-е изд. - СПб.: ПИТЕР, 2008. - 669 с.: ил. - (Учебник для вузов).
 5. Бобровский С. И. Delphi 7: учебный курс/ С. И. Бобровский. - Москва [и др.]: Питер, 2007. - 736 с.: ил. - (Учебный курс).
 6. Гордеев А. В. Операционные системы: учебник для студентов вузов/ А. В. Гордеев. - 2-е изд. - СПб.: ПИТЕР, 2007. - 416 с.: ил. - (Учебник для вузов).
 7. Сеннов А. С. Access 2007: учебный курс/ А. С. Сеннов. - Москва [и др.]: Питер, 2008. - 267 с.: ил. + 1 эл. опт. диск (CD-ROM). - (Учебный курс).
 8. Базы данных: учебник для вузов/ А. Д. Хомоненко, В. М. Цыганков, М. Г. Мальцев ; под ред. А. Д. Хомоненко. - 5-е изд., доп. - СПб.: КОРОНА принт; М.: Бином-Пресс, 2006. - 736 с.: ил.
- ### **8.3. Интернет ресурсы**

<http://www.cplusplus.com/doc/tutorial/>
<http://citforum.ru/programming/c.shtml>

9. Материально-техническое обеспечение дисциплины (модуля):

1. Материально-техническое обеспечение дисциплины включает:
2. Компьютерный класс, оборудованный для проведения лекционных и практических занятий средствами оргтехники, персональными компьютерами, объединенными в сеть с выходом в Интернет;
3. Установленное лицензионное программное обеспечение;
4. Библиотечный фонд РФ ПГУ им. Т.Г. Шевченко;
5. Мультимедийное оборудование для чтения лекций-презентаций;
6. Пакет демонстрационных программ.

10. Методические рекомендации по организации изучения дисциплины:

Рабочая учебная программа по дисциплине «Программирование и основы алгоритмизации» составлена в соответствии с требованиями Федерального Государственного образовательного стандарта ВО по направлению 15.03.04 «Автоматизация технологических процессов и производств».

Изучение дисциплины проходит в форме лекционных занятий, выполнения практических работ в лекционной/компьютерной аудитории. Самостоятельная работа заключается в самостоятельном изучении тем студентом, а так же в конспектировании тем и написании по ним эссе, рефератов.

Рубежный контроль - **курсовая работа и экзамен.**

Итоговый контроль – **экзамен.**

11. Технологическая карта дисциплины

Курс 1,2 семестр

Преподаватель – лектор **Корлюга Богдан Константинович**

Преподаватели, ведущие практические занятия **Корлюга Богдан Константинович**

Кафедра **АТПП**

Весовой коэффициент дисциплины в совокупной рейтинговой оценке, рассчитываемой по всем дисциплинам (если введена модульно-рейтинговая система) модульно-рейтинговая система не введена

Наименование дисциплины / курса	Уровень/ступень образования (бакалавриат, специалитет, магистратура)	Статус дисциплины в рабочем учебном плане (А, Б, В, Г) (если введена модульно-рейтинговая система)	Количество зачетных единиц / кредитов	
Смежные дисциплины по учебному плану (перечислить):				
ВВОДНЫЙ МОДУЛЬ (входной рейтинг-контроль, проверка «остаточных» знаний по смежным дисциплинам)				
Тема, задание или мероприятие входного контроля	Виды текущей аттестации	Аудиторная или внеаудиторная	Минимальное количество баллов	Максимальное количество баллов
Итого:				

БАЗОВЫЙ МОДУЛЬ (проверка знаний и умений по дисциплине)				
Тема, задание или мероприятие текущего контроля	Виды текущей аттестации	Аудиторная или внеаудиторная	Минимальное количество баллов	Максимальное количество баллов
Итого:				
ДОПОЛНИТЕЛЬНЫЙ МОДУЛЬ				
Тема, задание или мероприятие дополнительного контроля	Виды текущей аттестации	Аудиторная или внеаудиторная	Минимальное количество баллов	Максимальное количество баллов
Или				
Итого максимум:				

Необходимый минимум для получения итоговой оценки или допуска к промежуточной аттестации _____ баллов *(если введена модульно-рейтинговая система)*.

Дополнительные требования для студентов, отсутствующих на занятиях по уважительной причине: *(например, устное собеседование с преподавателем по проблемам пропущенных практических занятий, обязательное выполнение внеаудиторных контрольных и письменных работ и т.д.)*.

Приложение:

МЕТОДИКА ЗАПОЛНЕНИЯ ТЕХНОЛОГИЧЕСКОЙ КАРТЫ ДИСЦИПЛИНЫ

Преподаватель самостоятельно устанавливает минимальное и максимальное количество баллов, которое может получить студент за весь период изучения дисциплины (сумма минимального количества баллов условно равна 30% освоения дисциплины, сумма максимального количества баллов условно равна 100% освоения дисциплины). Преподаватель устанавливает диапазон баллов в пределах весового коэффициента дисциплины (в зависимости от стоимости дисциплины в зачетных единицах).

В технологической карте дисциплины выделяются следующие **модули**:

Вводный модуль - что уже знает и умеет студент из ранее изученных дисциплин;

Базовый модуль - чему должен научиться студент;

Дополнительный модуль - что может сделать студент для повышения оценки; условия промежуточной аттестации (автоматическое выставление оценки по результатам текущей работы студента, собеседование, экзамен и др.).

Составитель Кару /Корлюга Б.К./, старший преподаватель

Зав. кафедрой (обслуживающей дисциплину) Федоров В.Е., доцент, к.э.н.

Согласовано:

Директор филиала ПГУ им. Т.Г.Шевченко
в г. Рыбница: профессор

Павлинов Игорь Алексеевич Павлинов Игорь
Алексеевич