

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ
КЫРГЫЗСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМ. И. АРАБАЕВА
ФАКУЛЬТЕТ ФИЗИКО-МАТЕМАТИЧЕСКОГО ОБРАЗОВАНИЯ И
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

«Утверждаю»

декан ФФМОиИТ

доц. Бексултанов Ж.Т.



09 2019 г.

РАБОЧАЯ ПРОГРАММА

По дисциплине: Программирование

Тип дисциплины: профессиональный, базовый

Направление подготовки: 710100 Информатика и вычислительная техника

Профиль подготовки: Автоматизированные системы обработки информации и управления

Рабочая программа составлена на основании ГОС ВПО и СПО КР утвержденного МО и Н КР приказом № 1178/1 от «15» 09 2015 г., пер № 1025/1 и учебного плана по данному направлению, утвержденному от «21» 04 2018 г.

Разработчики рабочей программы:

Ст. преп. Асанбекова Н.О.

Обсуждено:

Одобрено:

на заседании кафедры ПИ

Учебно-методическим советом ФФМОиИТ

Протокол № 4

Протокол № 1

«5» сентября

«6» сентября 2019 г.

Зав.кафедрой ПИ

д.ф.-м.н., проф., Бийбосунов Б.И.

Председатель УМС ФФМОиИТ

Курс-2

Семестр -3, 4

Количество учебных недель в семестре – 1 семестр – 16 недель, 2 семестр – 12 недель

Форма итогового контроля (экзамен) – экзамен, экзамен

Число кредитов – 3 семестр – 2 кредита, 4 семестр – 2 кредита.

Всего часов по учебному плану – 3 семестр – 60 часов, 4 семестр – 60 часов.

Всего часов по учебному плану	Количество академических часов			
	очная			
	лекция	практика (семинар)	лаб.раб	срс
3 семестр	34	-	26	60
4 семестр	32	-	28	60

Раздел 1. Общие положения

1.1 Аннотация дисциплины: Программирование является одним из фундаментальных, инструментальных методов современной информатики. Поэтому учебный курс программирования занимает одно из центральных мест в системе подготовки учителя информатики и имеет как мировоззренческое, так и прикладное значение.

Главная цель курса - формирование системы понятий, знаний, умений и навыков в области современного программирования.

Учебный материал дисциплины базируется на развитии содержательной линии алгоритмизации и программирования школьного курса информатики, основой которого является освоение учащимися начальных знаний, умений и навыков в области структурного подхода конструированию алгоритмов и способов их реализации, как правило, в рамках методов и средств процедурного программирования.

Актуальность дисциплины: нынешнее программирование многогранно и используется в таких важных сферах как строительство, бизнес и экономика, медицина, биология и физика. Большой процент физического труда в промышленности заменен на машинный и роботизированный труд, который управляется посредством программного обеспечения, что обеспечивает существенный прирост скорости, точности операций и эффективности производства. Такое богатство разнообразия применений обеспечивается солидным выбором языков программирования, у каждого из которых есть свои плюсы и минусы.

1.2. Компетенции, формируемые в результате освоения дисциплины:

а) универсальные:

- общенаучными (ОК):

- владеть целостной системой научных знаний об окружающем мире, способен ориентироваться в ценностях жизни, культуры (ОК-1);
- способен использовать базовые положения математических /естественных/ гуманитарных/ экономических наук при решении профессиональных задач (ОК-2);
- способен приобретать новые знания с большой степенью самостоятельности с использованием современных образовательных и информационных технологий (ОК-3);

- инструментальными (ИК):

- способен воспринимать, обобщать и анализировать информацию, ставить цели и выбирать пути ее достижения (ИК-1);
- способен логически верно, аргументировано и ясно строить свою устную и письменную речь на государственном и официальном языках (ИК-2);
- владеть основными методами, способами и средствами получения, хранения и переработки информации, навыками работы с компьютером, как средством управления информацией, в том числе в глобальных компьютерных сетях и корпоративных информационных системах (ИК-5);

- социально-личностными и общекультурными (СЛК):

- способен социально взаимодействовать на основе принятых в обществе моральных и правовых норм, проявлять уважение к людям, толерантность к другой культуре, готовность к поддержанию партнерских отношений (СЛК-1);
- умеет критически оценивать свои достоинства и недостатки, наметить пути и выбрать средства развития достоинств и устранения недостатков (СЛК-2);
- способен проявлять готовность к диалогу на основе ценностей гражданского демократического общества, способен занимать активную гражданскую позицию (СЛК-3);
- способен использовать полученные знания, необходимые для здорового образа жизни, охраны природы и рационального использования ресурсов (СЛК-4);
- способен работать в коллективе, в том числе над междисциплинарными проектами (СЛК-5).

б) профессиональные (ПК)

- способен проводить предпроектное обследование объекта проектирования, системный анализ предметной области, их взаимосвязей (ПК-1);
- способен проводить техническое проектирование (ПК-2);
- способен проводить рабочее проектирование (ПК-3);

- способен проводить выбор исходных данных для проектирования (ПК-4);
- способен проводить моделирование процессов и систем (ПК-5);
- способен оценивать надежность и качество функционирования объекта проектирования (ПК-6);
- способен проводить сбор, анализ научно-технической информации, отечественного и зарубежного опыта по тематике исследования, способен участвовать в постановке и проведении экспериментальных исследований (ПК-22);
- готов использовать математические методы обработки, анализа и синтеза результатов профессиональных исследований (ПК-24);

1.3. Цель преподавания дисциплины:

Цель данного курса - дать студентам достаточно полное и строгое представление о современных языках программирования (на примере языка Python), алгоритмах программирования.

1.4. Задачи преподавания дисциплины:

формирование у студентов минимально необходимых знаний по дисциплине;

- ознакомление с техническими, алгоритмическими, программными и технологическими решениями, используемыми в данной области;
- выработка практических навыков аналитического и экспериментального исследования основных методов и средств, используемых в области, изучаемой в рамках данной дисциплины.

В результате теоретического изучения дисциплины студент должен знать: лексику и структуру языка Python, основные принципы построения больших программ на языке Python.

В результате практического освоения дисциплины студент должен уметь создавать на языке Python программы различного типа, владеть диалоговым и графическим инструментарием ОС.

В ходе теоретического изучения и практического, лабораторного освоения дисциплины студент должен приобрести навыки: ввода и вывода данных в консольном режиме, с помощью инструментов оконного интерфейса и файлов; работы с графическим инструментарием программной среды; создания и обработки статических и динамических структур данных; создания и работы с типами данных, определяемых пользователем.

1.5. Взаимосвязь учебных дисциплин

Пререквизиты:

Для формирования общекультурных и профессиональных компетенций необходимы базовые компетенции, сформированные при изучении учебных дисциплин учебного плана «Информатика», «Архитектура вычислительных систем», «Математический анализ».

Для успешного изучения дисциплины студент должен:

знать основные принципы организации и функционирования современного компьютера; вид представления информации различного рода в памяти современного компьютера; позиционные системы счисления, способы перевода чисел из одной системы в другую; основы математического анализа;

уметь: применять математические методы и вычислительные алгоритмы для решения практических задач, проектировать эксперимент и анализировать результаты; производить переводы чисел из одной системы счисления в другую; выполнять арифметические операции над двоичными и шестнадцатеричными числами;

владеть: методами построения математической модели задач и содержательной интерпретации полученных результатов; основными методами, способами и средствами получения, хранения, переработки информации, навыками работы с компьютером как средством управления информацией.

Кроме того, знание английского языка облегчит усвоение среды программирования.

Постреквизиты:

Компетенции, сформированные в результате освоения содержания дисциплины «Программирование», необходимы для освоения большинства дисциплин профессионального цикла учебного плана.

В результате выполнения учебной программы курса студент должен

иметь представление:

- о структуре программного обеспечения, о видах и областях применения основных программных средств;
- об основных этапах решения задач на ЭВМ, о жизненном цикле программы, о критериях качества программы, о понятии диалоговых и дружественных пользователю программ;
- об общих принципах построения, описания, способах записи и основах доказательства правильности алгоритмов (верификации алгоритмов);
- об общей классификации современных языков программирования, их

областях применения и использования;

- о стандартных типах данных и о типах данных, определяемых пользователем;
- о представлении основных структур программирования: итерация, ветвление, повторение, процедуры и функции; о программирование рекурсивных алгоритмов;
- о файлах и об основных видах динамических структур данных и способах их реализации;
- о способах конструирования программ и о понятии модульных программ;
- об использовании дополнительных пакетов и библиотек при программировании;

знать:

- современные методы и средства разработки алгоритмов и программ на языке Python;
- синтаксис и семантику основных конструкций языка Python;
- способы организации сложных структур данных (массивы, структуры, списки, деревья), основные методы представления и алгоритмы обработки этих данных;

уметь:

- принимать участие во всех фазах проектирования, разработки, изготовления, отладки и документирования программ;
- разрабатывать модульных программ на Паскаль и Delphi с использованием базовых типов данных и массивов и сложных иерархических типов данных и переменных;
- разрабатывать собственные программы с использованием стандартных фрагментов алгоритмов;

получить навыки:

- разработки программ, использующих данные в произвольном формате, а также использующих массивы указателей и списки для хранения, упорядочения и поиска данных;
- проектирования программ, использующих двоичные файлы для размещения различных структур данных с полной и поэлементной их загрузкой в память.

Раздел 2. Содержание дисциплины и формируемые компетенции

№ темы	Наименование тем и разделов по дисциплине	Вид контроля	Компетенции
III семестр			

1 модуль		Текущий контроль	
1	Введение. Знакомство со средой Python. Алфавит языка.	Контрольная работа, индивидуальные задания, тестирование	ОК-1, ОК-2,
2	Структура программы.		ОК-3, ИК-1,
3	Выражения и операции. Стандартные функции. Примеры.		ИК-2, ИК-5,
4	Операторы языка. Программирование алгоритмов линейной структуры. Примеры.	Контрольная работа, индивидуальные задания, тестирование	СЛК-1, СЛК-2, СЛК-3, СЛК-4, СЛК-5, ПК-1, ПК-2, ПК-3, ПК-4, ПК-5, ПК-6, ПК-22, ПК-24
2 модуль		Текущий контроль	
1	Программирование алгоритмов разветвляющей структуры. Оператор if, elif, else	Контрольная работа, индивидуальные задания, тестирование	ОК-1, ОК-2, ОК-3, ИК-1, ИК-2, ИК-5,
2	Программирование алгоритмов циклической структуры: структура цикла “ while ”.		СЛК-1, СЛК-2, СЛК-3, СЛК-4,
3	Программирование алгоритмов циклической структуры: структура цикла “ for ”.	Контрольная работа, индивидуальные задания, тестирование	СЛК-5, ПК-1, ПК-2, ПК-3, ПК-4, ПК-5, ПК-6, ПК-22, ПК-24
Итоговый (семестровый) контроль:		Экзамен	
IV семестр			
1 модуль		Текущий контроль	
1.	Массивы. Модуль array	Контрольная работа, индивидуальные задания, тестирование	ОК-1, ОК-2,
2.	Одномерный массив		ОК-3, ИК-1,
3.	Двумерный массив		ИК-2, ИК-5,
4.	Процедуры.		СЛК-1, СЛК-2,
2 модуль		Текущий контроль	
5.	Функции	Контрольная работа, индивидуальные задания, тестирование	СЛК-3, СЛК-4, СЛК-5, ПК-1,
6.	Графика. Модуль graphics		ПК-2, ПК-3,
7.	Графика. Модули turtle, Matplotlib		ПК-4, ПК-5, ПК-6, ПК-22,

			ПК-24
--	--	--	-------

Общая трудоемкость дисциплины в семестре по реализуемым формам обучения

№ п/п	Порядковый номер темы дисциплины (Тема №)	Количество академических часов			
		Очная			
		лекция	практика (семинар)	лаб. раб.	срс
III семестр (34/26)					
1	Введение. Знакомство со средой Python. Алфавит языка.	2			
2	Структура программы.	4		2	
3	Выражения и операции. Стандартные функции. Примеры.	4		2	
4	Операторы языка. Программирование алгоритмов линейной структуры. Примеры.	4		4	
5	Программирование алгоритмов разветвляющей структуры. Оператор if, elif, else	4		6	
6	Программирование алгоритмов циклической структуры: структура цикла "while".	8		6	
7	Программирование алгоритмов циклической структуры: структура цикла "for".	8		6	
Общий объем учебной нагрузки (в часах)		34		26	
Всего часов:		34		26	
IV семестр (32/28)					
1.	Массивы. Модуль array	2		2	
2.	Одномерный массив	4			
3.	Двумерный массив	6		6	
4.	Процедуры.	4		4	
5.	Функции	4		4	
6.	Графика. Модуль graphics	6		6	
7.	Графика. Модули turtle, Matplotlib	6		6	
Общий объем учебной нагрузки (в часах)		32		28	
Всего часов:		32		28	

Раздел 3. Структура и содержание лабораторных занятий, практических (семинарских) и СРС

3.2. Лабораторные занятия

№ практ. (сем.) занятия	наименование и краткое содержание занятия	характер и цель занятия, формируемые компетенции	кол-во баллов
III семестр (26)			
1	Структура программы.	Научить стандартным функциям и основным операциям. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК - 1,2,3,4,5,6,22,24	2
2	Выражения и операции. Стандартные функции. Примеры.	Научить стандартным функциям и основным операциям. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК - 1,2,3,4,5,6,22,24	2
3	Операторы языка. Программирование алгоритмов линейной структуры. Примеры.	Познакомить линейными программами. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК - 1,2,3,4,5,6,22,24	4
4	Программирование алгоритмов разветвляющей структуры. Оператор if, elif, else	Научить правильно сформировать условия и решение задач с разветвлением. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК - 1,2,3,4,5,6,22,24	6
5	Программирование алгоритмов циклической структуры: структура цикла "while".	Ознакомить с циклами. Научить составить циклические программы с предусловием. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК - 1,2,3,4,5,6,22,24	6
6	Программирование алгоритмов циклической структуры: структура цикла "for".	Ознакомить с циклами. Научить составить циклические программы с параметром ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК - 1,2,3,4,5,6,22,24	6
IV семестр (28)			
1	Массивы. Модуль array	Дать основное определение и понятие о массивах. Их видах. Разобрать на примерах одномерные массивы. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК - 1,2,3,4,5,6,22,24	2
2	Одномерный массив		
3	Двумерный массив	Ознакомиться с двумерными массивами.	6

		Решить задачи на двумерные массивы. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК -1,2,3,4,5,6,22,24	
4	Процедуры.	Ознакомить студентов с процедурами и подпрограммами. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК -1,2,3,4,5,6,22,24	4
5	Функции	Ознакомить студентов с функциями, и показать различия между подпрограммами. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК -1,2,3,4,5,6,22,24	4
6	Графика. Модуль graphics	Ознакомить студентов с графическим редактором. Рассмотреть на примерах все возможности графического модуля. ОК-1,2,3 ИК-1,2,5, СЛК-1,2,3,4,5 ПК -1,2,3,4,5,6,22,24	6
7	Графика. Модули turtle, Matplotlib		6

3.3. Структура СРС (примерная)

№ п/п	порядковый номер темы дисциплины (Тема №)	форма отчетности (предлагаемые)	кол-во баллов
III семестр			
1	Введение. Знакомство со средой Python. Алфавит языка.	Доклад	2
2	Структура программы.	Доклад	2
3	Выражения и операции. Стандартные функции. Примеры.	Презентация	2
4	Операторы языка. Программирование алгоритмов линейной структуры. Примеры.	Презентация	2
5	Программирование алгоритмов разветвляющей структуры. Оператор if, elif, else	Презентация с программой	2
6	Программирование алгоритмов циклической структуры: структура цикла “ while”.	Презентация с программой	2
7	Программирование алгоритмов циклической структуры: структура цикла “ for”.	Презентация с программой	2
IV семестр			

1	Массивы. Модуль array	Доклад	2
2	Одномерный массив	Презентация	2
3	Двумерный массив	Презентация	2
4	Процедуры.	Доклад	2
5	Функции	Презентация с программой	2
6	Графика. Модуль graphics	Презентация с программой	2
7	Графика. Модули turtle, Matplotlib	Презентация с программой	2

Раздел 4. Образовательные технологии

Проектное обучение.

Целевое назначение: самостоятельный мотивированный поиск решения познавательных и практических задач, способствующий развитию мышления.

Задачи: пробуждение, развитие культивирование проектного, т.е. проблемно-поискового, конструктивно-волевого, критического мышления.

Педагогический инструментарий: в качестве инструментария используется консультационный, диалоговый тип работы, в результате которой преподаватель стимулирует мотивацию студентов на успех.

Ключевые умения и компетентности: исследовательские, социальные, информационные, познавательные, проектировочные.

Модульное обучение.

Целевое назначение: организация самостоятельного учения студентов, удовлетворение образовательных потребностей каждого студента в соответствии с его склонностями, интересами, возможностями, мотивационное управление учением, развитие интеллекта, самостоятельности, коллективизма, умения осуществлять самоуправление учебно-познавательной деятельностью.

Задачи: Позволяет наиболее эффективно реализовывать организационную, координационную, интегративную функции будущего специалиста и модели социального партнерства в обществе.

Педагогический инструментарий: Включает работу в парах и в группах с консультантом, самостоятельная и фронтальная работа, работа с картографическим материалом, с техническими средствами обучения, с таблицами и иллюстрациями и многое другое.

Ключевые умения и компетентности: развитие мотивационной сферы студента, интеллекта, самостоятельности, коллективизма, склонностей, умений осуществлять самоуправление учебно-познавательной деятельностью.

Кейсовое обучение.

Целевое назначение: Развитие аналитических, исследовательских, коммуникативных навыков, выработка умения анализировать ситуацию, планировать стратегию и принимать управленческие решения в процессе нахождения плана действий в специально созданной преподавателем ситуации.

Задачи: Посредством эффективно выстроенной системы кейсов происходит непосредственное изучение, анализ и интерпретация деловых ситуаций или задач.

Педагогический инструментарий: Методы группового взаимодействия: споры и дискуссии.

Ключевые умения и компетентности: развитие способности к профессиональной ориентации через формирование устойчивых компетентностных структур.

Портфолио.

Целевое назначение: Детализировать процесс образовательного участия каждого студента, увидеть картину значимых образовательных результатов, в целом обеспечить отслеживание индивидуального прогресса учащегося в широком образовательном контексте, продемонстрировать его способность практически применять приобретенные знания и умения.

Задачи: Поддержание на высоком уровне учебной мотивации, развитие активности поискового характера для решения индивидуальных задач, саморазвитие студентов, развитие рефлексии и самооценки, умение структуризировать собственную образовательную траекторию развития (цели, задачи, результат).

Педагогический инструментарий: Проектные и исследовательские работы, самостоятельное творчество, самомотивация и самосовершенствование.

Ключевые умения и компетентности: повышение образовательной активности студентов, уровня осознания ими своих целей и возможностей, что позволяет максимально соотнести результат индивидуального образовательного маршрута с выбранной профессиональной деятельностью.

Интерактивное обучение.

Целевое назначение: Активизация индивидуальных, умственных процессов обучающихся, возбуждение внутреннего диалога студента, обеспечение пониманию информации, являющейся предметом обмена, индивидуализация педагогического взаимодействия, вывод ученика на позицию субъекта обучения, достижение взаимной двухсторонней связи преподавателя и студента.

Задачи: фасилитация (поддержка, облегчение) – направление и помощь процессу обмена информацией, выявление многообразия точек зрения, обращение к личному опыту студентов, поддержка активности участников, соединение теории и практики, обогащение опыта, облегчение восприятия.

Педагогический инструментарий: развитие диалогового общения, которое ведет к взаимопониманию, взаимодействию, к совместному решению общих, но значимых для каждого участника задач, участие в дискуссиях, общение с другими людьми.

Ключевые умения и компетентности: в ходе диалогового обучения студенты учатся критически мыслить, решать сложные проблемы на основе анализа обстоятельств и соответствующей информации, взвешивать альтернативные мнения, принимать продуманные решения.

Раздел 5. Процедура оценки достижений студентов

Перевод 100-балльной рейтинговой оценки по дисциплине в традиционную четырех балльную

100-балльная система оценки	Традиционная четырехбалльная система оценки
85 – 100 баллов	оценка «отлично»/«зачтено»
70 – 84 баллов	оценка «хорошо»/«зачтено»
50 – 69 баллов	оценка «удовлетворительно»/«зачтено»
менее 50 баллов	оценка «неудовлетворительно»/«незачтено»

Распределение баллов, формирующих рейтинговую оценку работы студента, осуществляется следующим образом:

Виды работ	Максимальное количество баллов
I, II модули	
Посещаемость	20
Текущий контроль	20
Творческий рейтинг	20
Промежуточная аттестация (тестирование)	40
ИТОГО	100
Итоговая оценка	
СРС	50
Текущий контроль	50
ИТОГО	100

Контрольные вопросы:

Знакомство со средой Python. Алфавит языка. Идентификаторы. Служебные слова. Структура программы. Название программы. Разделы описания переменных, констант, функций, процедур, меток, тело программы.

Стандартные математические функции. Правила их написания.

Математические и логические выражения.

Программирование линейной структуры. Какие программы называются линейными?

Программирование разветвляющей структуры. Условный оператор if..then..else.

Оператор выбора.

Программирование алгоритмов циклической структуры. Цикл «с параметром», цикл «пока», цикл «до». Вложенные циклы.

Одномерные массивы, двумерные массивы.

Процедуры и функции. Процедуры с параметром. Процедуры без параметра.

Символьные и строковые переменные.

Графический модуль.

Раздел 6. Средства и материально-техническое обеспечение дисциплины

Реализация программы дисциплины требует наличия учебного кабинета теоретического обучения; лаборатории системного и прикладного программирования.

Оборудование учебного кабинета: посадочные места по количеству обучающихся, рабочее место преподавателя, комплект учебно-методической документации.

Технические средства обучения: ноутбук, экран, мультимедийный проектор, доска.

Оборудование лаборатории и рабочих мест лаборатории: персональные компьютеры (монитор, системный блок, клавиатура, мышь), комплект учебно-методической документации, программное обеспечение (оболочки языков программирования).

Раздел 7. Учебно-методическое и информационное обеспечение дисциплины

7.1. В обучении студентов применяются методы получения нового знания (лекция-визуализация, лекция-беседа, лекция-дискуссия, тематические и проблемные семинары, лабораторные и практические занятия и т.д.), проверки знаний (устный опрос, контрольные работы, тестирование и т.д.) методы формирования умений, навыков и самостоятельной работы.

7.2. Электронные курсы

<https://stepik.org/course/3352/promo>

<https://vse-kursy.com/onlain/676-programmirovanie-na-vazyke-pascal.html>

<http://kurs-pascal.narod.ru/>

<https://www.intuit.ru/studies/courses/41/41/info>

<https://canvas.instructure.com/courses/928819/pages/vviedieniie-v-programmirovaniie-paskal>

7.3. Рекомендуемая литература

Основная литература

1. Доусон М. Програмуем на Python. – СПб.: Питер, 2014. – 416 с.
2. Лутц М. Изучаем Python, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 1280 с.
3. Лутц М. Программирование на Python, том I, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с.
4. Лутц М. Программирование на Python, том II, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с.
5. Прохоренок Н.А. Python 3 и PyQt. Разработка приложений. – СПб.: БХВ-Петербург, 2012. – 704 с.
6. Вабищевич П. Н. Численные методы. Вычислительный практикум. – – 320 с.
7. Пилгрим Марк. Погружение в Python 3 (Dive into Python 3 на русском)
8. Прохоренок Н.А. Самое необходимое. — СПб.: БХВ-Петербург, 2011. — 416 с.
9. Хахаев И.А. Практикум по алгоритмизации и программированию на Python. – М.: Альт Линукс, 2010. — 126 с. (Библиотека ALT Linux).

Дополнительные литературы

1. Чаплыгин А.Н. Учимся программировать вместе с питоном.
2. Шапошникова С. Основы программирования на Python. Вводный курс.
3. Briggs J. R. — Python for Kids — 2012
4. Deitel Н.М. et al. Python — How to Program
5. Allen Downey – ThinkPython+Kart[Python_3.2]
6. Абрамов С. А., Гнезделова Г.Г., Капустина Е.Н., Селен М.И. Задачи по программированию. Москва: Наука, 1988.
7. Шестакович В.П. «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ» Минск 2010.