

МИНИСТЕРСТВО ОБЩЕГО И ПРОФЕССИОНАЛЬНОГО
ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ
Государственное автономное образовательное учреждение дополнительного
профессионального образования Свердловской области
«ИНСТИТУТ РАЗВИТИЯ ОБРАЗОВАНИЯ»
(ГАОУ ДПО СО «ИРО»)
Кафедра информационных технологий

Утверждено
Научно-методическим советом
ГАОУ ДПО СО «ИРО»
протокол № 1 от 21.02.2019 г.
Утверждено
Экспертным советом
ГАОУ ДПО СО «ИРО»
протокол № 1 от 21 января 2019 г.

Дополнительная профессиональная программа
повышения квалификации

**ПОДГОТОВКА К ЕГЭ ПО ИНФОРМАТИКЕ:
МЕТОДИКА ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ.
ВАРИАТИВНЫЙ МОДУЛЬ
«ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON»**

(40 час.)

Автор:
Долинер Л.И., проф. кафедры
информационных технологий
ГАОУ ДПО СО «ИРО»,

Екатеринбург
2019

ОГЛАВЛЕНИЕ

АННОТАЦИЯ.....	3
ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.....	4
Раздел 1. ХАРАКТЕРИСТИКА ПРОГРАММЫ	6
1.1. Цель реализации программы.....	6
1.2. Планируемые результаты обучения	6
1.3. Категория обучающихся	6
1.4. Форма обучения.....	6
1.5. Срок освоения программы, режим занятий	6
Раздел 2. СОДЕРЖАНИЕ ПРОГРАММЫ.....	8
2.1. Учебные планы	
2.1.1. Учебно-тематический план: очная форма обучения	8
2.1.2. Учебно-тематический план: очно-заочная форма обучения с использованием ДОТ	9
2.1.3. Учебно-тематический план: заочная форма обучения с использованием ДОТ	11
2.2. Сетевая форма.....	13
2.3. Рабочая программа	14
2.3.1. Рабочая программ: очная форма обучения.....	14
2.3.2. Рабочая программа: очно-заочная форма обучения	17
2.3.3. Рабочая программа: заочная форма обучения с использованием ДОТ21	
Раздел 3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК	25
Раздел 4. ФОРМЫ АТТЕСТАЦИИ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ	25
4.1. Входной контроль	Ошибка! Закладка не определена.
4.2. Текущий контроль	Ошибка! Закладка не определена.
4.3. Итоговая аттестация.....	Ошибка! Закладка не определена.
Литература	Ошибка! Закладка не определена.

АННОТАЦИЯ

Дополнительная профессиональная программа повышения квалификации «Подготовка к ЕГЭ по информатике: Методика обучения программированию: Вариативный модуль: «Основы программирования на языке Python»» (40 час.) адресована педагогическим работникам, реализующим образовательные программы по информатике и информационным и коммуникационным технологии (ИКТ). Кроме того, данная ДПП будет полезна тем учителям информатики, которые являются экспертам, участвующим в проверке выполнения открытой части ЕГЭ по информатике.

Цель вариативного модуля: Совершенствование профессиональных компетенций слушателей в области алгоритмизации и программирования на языке Python и методики преподавания программирования для выполнения трудовой функции «Предметное обучение. Математика» согласно профессиональному стандарту «Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспитатель, учитель)».

Категория слушателей: учителя информатики.

Возможные формы обучения слушателей: очная, очно-заочная с использованием дистанционных образовательных технологий и полностью заочная (также с использованием дистанционных образовательных технологий) при обеспечении соответствующих необходимых технических и технологических условий.

Продолжительность обучения 40 часов.

Итоговая аттестация слушателей проводится в виде контрольной работы по решению задач ЕГЭ по программированию.

По результатам освоения дополнительной профессиональной программы и успешного прохождения итоговой аттестации слушателям выдается удостоверение о повышении квалификации.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная профессиональная программа повышения квалификации «Подготовка к ЕГЭ по информатике: Методика обучения программированию: Вариативный модуль: «Основы программирования на языке Python»» (40 час.) адресован педагогическим работникам, реализующим образовательные программы по информатике, а также экспертам, участвующим в проверке выполнения открытой части ЕГЭ по информатике.

В эпоху стремительно развивающихся информационных технологий обучающиеся должны быть не только информационно грамотными, но и иметь навыки управления информационными системами, а, значит, основы алгоритмизации и программирования становятся неотъемлемой частью среднего образования. В связи с этим, обучающиеся, ориентированные на дальнейшее обучение в сфере IT, заинтересованы в изучении основ программирования на разных программных платформах. В образовательных организациях Свердловской области наблюдается ситуация, когда учителя информатики владеют каким-либо одним языком программирования – Бейсик или Паскаль. Эти языки позволяют познакомить учащихся с основами программирования, но они устарели, а разработать с помощью этих языков проект современного уровня весьма проблематично.

Существует и проблема экспертизы открытой части заданий ЕГЭ, где обучающиеся имеют право создавать программы на любом известном им языке программирования, а эксперт предметной комиссии обязан эту работу проверить.

Перечисленные аргументы стали условием для создания и последующей реализации данного вариативного модуля: существует необходимость подготовки учителей информатики в области изучения основ современных языков программирования и методики их преподавания.

Результаты ЕГЭ 2017-2018 гг. показали, что часть выпускников школ выбирают для решения задач язык программирования Python. И это уже не единичные случаи.

Цель вариативного модуля: Совершенствование профессиональных компетенций слушателей в области алгоритмизации и программирования на языке Python и методики преподавания программирования для выполнения трудовой функции «Предметное обучение. Математика» согласно профессиональному стандарту «Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспитатель, учитель)».

Категория слушателей: учителя информатики и ИКТ.

Для успешной организации практических занятий и самостоятельной работы слушателей предусматривается активное использование комплекта учебно-методического обеспечения.

Формы учебной работы: лекции, лабораторные и практические занятия, консультации.

Каждый раздел программы предполагает обязательное время на индивидуальную или групповую рефлексия, осуществляемую в форме круглого стола; для преподавателя итоги рефлексии группы становятся основой для корректировки методов обучения и дают возможность осуществления обратной связи с обучающимися.

По результатам освоения дополнительной профессиональной программы и успешного прохождения итоговой аттестации слушателям выдается удостоверение о повышении квалификации государственного образца.

Методы обучения: традиционные, активные, интерактивные, с элементами дифференцированного обучения.

Возможные формы обучения слушателей: очная, очно-заочная с использованием дистанционных образовательных технологий и полностью заочная (также с использованием дистанционных образовательных технологий) при обеспечении соответствующих необходимых технических и технологических условий.

Объём программы составляет 40 академических часов.

Срок обучения:

Очная форма обучения:

5 дней, 8 часов в день с 9:00 до 16:10 ежедневно

Очно-заочная форма обучения (с использованием ДОТ):

Очно, 16 часов:

первый день: установочное занятие, с 9:00 до 16:10

последний день: 4 часа: анализ выполненной работы и консультация; 4 часа – итоговая аттестация.

Заочно, 24 часа:

10 дней между первым и последним очными днями; приблизительное время работы – до 1,5 часов ежедневно

Заочная форма обучения (с использованием ДОТ)

Срок освоения – 21 день. Режим занятий определяется слушателями индивидуально.

Формы итоговой аттестации: выполнение контрольной работы (решение задач ЕГЭ по программированию на языке Python).

Достижение поставленной цели и соответствие уровня подготовки обучающихся требованиям выявляется в практической деятельности в ходе реализации программы, итоговой аттестации.

Раздел 1. ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1. Цель реализации программы

Совершенствование профессиональных компетенций слушателей в области алгоритмизации и программирования на языке Python и методики преподавания программирования для выполнения трудовой функции «Предметное обучение. Математика» согласно профессиональному стандарту «Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспитатель, учитель)».

1.2. Планируемые результаты обучения

Трудовые действия	Знания и умения
1. Формирование конкретных знаний, умений и навыков в области математики и информатики	Знать: • основные синтаксические конструкции и типы данных языка программирования Python; • формы и методы проверки и оценки результатов обучения программированию
	Уметь: • создавать программы на изучаемом языке программирования, обеспечивающие решение всех типов задач, которые встречаются в ЕГЭ по информатике; • понимать и анализировать программы, написанные на изучаемом языке; • выбирать эффективные методические приемы обучения основам алгоритмизации и программирования с учетом особенностей языка.

1.3. Категория обучающихся

Учителя информатики.

1.4. Форма обучения

Очная, очно-заочная (с использованием дистанционных образовательных технологий, ДОТ) и заочная (с использованием ДОТ) при обеспечении соответствующих необходимых технических и технологических условий.

1.5. Срок освоения программы, режим занятий

Программа рассчитана на 40 академических часов.

Очная форма обучения:

5 дней, 8 часов в день с 9:00 до 16:10 ежедневно

Очно-заочная форма обучения (с использованием ДОТ):

Очно, 16 часов:

первый день: установочное занятие, с 9:00 до 16:10

последний день: 4 часа: анализ выполненной работы и консультация; 4 часа – итоговая аттестация.

Заочно, 24 часа:

10 дней между первым и последним очными днями; приблизительное время работы – до 1,5 часов ежедневно

Заочная форма обучения (с использованием ДОТ)

Срок освоения – 21 день. Режим занятий определяется слушателями индивидуально.

Раздел 2. СОДЕРЖАНИЕ ПРОГРАММЫ

2.1. Учебные планы

2.1.1. Учебный (тематический) план

дополнительной профессиональной программы повышения квалификации

**ПОДГОТОВКА К ЕГЭ ПО ИНФОРМАТИКЕ:
МЕТОДИКА ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ.
ВАРИАТИВНЫЙ МОДУЛЬ
«ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON»
(40 час.)**

очная форма обучения

№ п/п	Наименование разделов (модулей) и тем	Всего, час.	Аудиторные учебные занятия. Учебные работы		Внеаудиторная работа С/р	Форма контроля
			Лекции	Практические и др. формы занятий		
1.	Тема 1. Особенности изучения программирования на языке Python	2	1	1		
2.	Тема 2. Ветвление	2	1	1		Компьютерный тест
3.	Тема 3. Реализация циклических алгоритмов	6	1	5		Выполнение практических заданий
4.	Тема 4. Функции	3	0	3		Выполнение практических заданий
5.	Тема 5. Алгоритмы работы со строками	6	1	5		Выполнение практических заданий
6.	Тема 6. Особенности изучения работы со списками и кортежами	6	1	5		Выполнение практических заданий
7.	Тема 7. Словари	4	1	3		Выполнение практических заданий
8.	Тема 8. Файлы	4	1	3		Выполнение практических заданий

9.	Тема 9. Решение задач ЕГЭ по программированию	3	0	3		
10.	Итоговая аттестация	4	0	4		Решение задач ЕГЭ по информатике (на программирование)
Итого:		40	7	33		

2.1.2. Учебный (тематический) план
 дополнительной профессиональной программы повышения квалификации
ПОДГОТОВКА К ЕГЭ ПО ИНФОРМАТИКЕ:
МЕТОДИКА ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ.
ВАРИАТИВНЫЙ МОДУЛЬ
«ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON»
(40 час.)

очно-заочная форма обучения с использованием ДОТ

№ п/п	Наименование разделов (модулей) и тем	Всего, час.	Аудиторные учебные занятия. Учебные работы		Внеаудиторная работа С/р	Форма контроля
			Лекции	Практические и др. формы занятий		
1.	Тема 1. Особенности изучения программирования на языке Python	2	1	1		
2.	Тема 2. Ветвление	2	1	1		
3.	Тема 3. Реализация циклических алгоритмов	6	1	3	2	Компьютерный тест
4.	Тема 4. Функции	3	0	0	3	Контрольная работа №1
5.	Тема 5. Алгоритмы работы со строками	6	0	0	6	
6.	Тема 6. Особенности изучения работы со списками и кортежами	6	0	0	6	
7.	Тема 7. Словари	4	0	0	4	
8.	Тема 8. Файлы	3	0	0	3	Контрольная работа №2
9.	Тема 8. Файлы	1	0	1	0	
10.	Тема 9. Решение задач ЕГЭ по программированию	3	0	3	0	

11.	Итоговая аттестация	4	0	4	0	Решение задач ЕГЭ по информатике (на программирование)
Итого:		40	3	13	24	

2.1.3. Учебный (тематический) план
 дополнительной профессиональной программы повышения квалификации
ПОДГОТОВКА К ЕГЭ ПО ИНФОРМАТИКЕ:
МЕТОДИКА ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ:
ВАРИАТИВНЫЙ МОДУЛЬ:
ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON
 (40 час.)

заочная форма обучения с использованием ДОТ

№ п/п	Наименование разделов (модулей) и тем	Всего, час.	Аудиторные учебные занятия. Учебные работы		Внеаудиторная работа С/р	Форма контроля
			Лекции	Практические и др. формы занятий		
1.	Тема 1. Особенности изучения программирования на языке Python	2			2	
2.	Тема 2. Ветвление	2			2	
3.	Тема 3. Реализация циклических алгоритмов	4			4	Компьютерный тест
4.	Тема 4. Функции	2	0	0	2	
5.	Тема 5. Алгоритмы работы со строками	3	0	0	3	Контрольная работа №1
6.	Тема 6. Особенности изучения работы со списками и кортежами	6	0	0	6	
7.	Тема 7. Словари	6	0	0	6	
8.	Тема 8. Файлы	4	0	0	4	Контрольная работа №2
9.	Тема 9. Решение задач ЕГЭ по программированию	3	0	0	3	

10.	Итоговая аттестация	4	0	0	4	Решение задач ЕГЭ по информатике (на программирование)
Итого:		40	0	0	40	

2.2. Сетевая форма Сетевая форма не предусмотрена

2.3. Рабочая программа

2.3.1. Рабочая программ: очная форма обучения

дополнительной профессиональной программы повышения квалификации

ПОДГОТОВКА К ЕГЭ ПО ИНФОРМАТИКЕ: МЕТОДИКА ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ: ВАРИАТИВНЫЙ МОДУЛЬ: ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON (40 час.)

№ п/п	Виды учебных занятий, учебных работ	Содержание	Знания/умения
Тема 1. Особенности изучения программирования на языке Python	<i>Вводная лекция</i>	Особенности языка программирования Python. Версии языка Python. Состав среды программирования. Организация рабочего пространства обучающегося. Особенности обучения языку программирования Python..	Знать базовые характеристики и особенности языка программирования Python; отличия версий 2.x и 3.x. Знать , где взять среду программирования Python. Знать средства и методы обучения программирования на языке Python, Уметь запускать среду IDLE.
	<i>Практическое занятие</i>	Основы работы с интерпретатором Python. Арифметические операции. Простейшие типы данных. Технология создания программы в Python. Функции ввода и вывода. Основы компьютерной графики.	Знать арифметические операции. Знать особенности использования операции присваивания в Python. Знать особенности использования функций ввода и вывода. Знать простейшие типы данных, используемых в Python. Уметь использовать интерпретатор для простейших вычисления. Уметь создавать простейшие линейные программы с использованием оператора присваивания, арифметических операций, функций ввода-вывода. Уметь создавать простейшие изображения: круг, эллипс, прямоугольник, линию и многоугольник.
Тема 2. Ветвление.	<i>Интерактивная лекция</i>	Основы алгоритмизации: понятие блок-схемы. Полное и неполное ветвление. Особенности решения задачи и оформление программ с	Знать основные элементы для составления блок-схем. Знать блок-схему для оператора ветвления. Знать назначения операторы AND, OR, NOT.

		конструкцией ветвления в Python. Сложные условия. Методика обучения, прогнозирование типичных ошибок обучающихся.	Знать основные приемы преподавания темы/
	Лабораторная работа	Разработка программ с использованием оператора ветвления, содержащего простые и сложные условия, вложенные ветвления, каскадные условия.	Уметь разрабатывать программы, содержащие оператор ветвления. Уметь разрабатывать программы, содержащие вложенные ветвления. Уметь разрабатывать программы, содержащие каскадные и сложные условия.
Тема 3. Реализация циклических алгоритмов	Интерактивная лекция	Циклы на языке Python: цикл for. Циклы с условиями. Типовые алгоритмы. Особенности преподавания темы, вариативность решения задач, подбор задач.	Знать , когда применяется цикл For. Знать , когда используются циклы с условиями. Знать , как проектируются условия для этих циклов.
	Лабораторная работа	Типовые задачи, реализуемые с помощью цикла for. Типовые задачи, реализуемые с помощью цикла while. Типовые алгоритмы: вычисление суммы, определение экстремума, вычисление суммы ряда.	Уметь решать задачи с использованием цикла For. Уметь реализовывать типовые алгоритмы: вычисление суммы и экстремума. Уметь проектировать условия для цикла For. Уметь решать задачи, в которых используются циклы с условием. Уметь проектировать условия для этих циклов. Уметь реализовывать типовые алгоритмы: вычисление суммы ряда с заданной точностью.
Тема 4. Функции	Лабораторная работа	Функции. Рекурсивные алгоритмы на языке Python. Локальные и глобальные переменные. Датчик случайных чисел. Методические приемы изучения темы	Знать , как оформляется функция в Python. Уметь выделять часть алгоритма, который целесообразно оформлять в виде функции. Понимать , как работает рекурсивный алгоритм. Уметь решать задачи, в которых используются рекурсивный алгоритм. Понимать , как работает датчик случайных чисел. Уметь использовать датчик случайных чисел для решения задач.
Тема 5.	Интерактивная	Строковый тип данных.	Знать , как задается строковая

Алгоритмы работы со строками	лекция	Срезы. Методы работы со строками. Методические аспекты преподавания	константа в Python. Знать , как делать срезы в Python. Знать методические. подходы изучения темы.
	Лабораторная работа	Как задать строковые константы. Срезы. Базовые функции: конкатенация, дублирование, определение длины строки. Типовые методы и алгоритмы: замена букв/фрагментов в строке; методы поиска, разбиения строк,	Уметь задавать строковые константы. Уметь делать срезы строк. Уметь использовать базовые функции работы со строками для решения задач. Уметь решать задачи с использованием типовых методов и алгоритмов: замена букв/фрагментов в строке; методы поиска, разбиения строк,
Тема 6. Особенност и изучения работы со списками и кортежами	Интерактивная лекция	Операции со списками. Генерация списков. Срезы. Методы сортировки списков. Вложенные списки. Матрицы. Кортежи.	Знать , что такое список. Знать , как создается список. Знать , как делать срезы списков. Знать основные методы и функции, используемые для работы со списками. Знать , что такое кортеж. Знать , как создается кортеж. Знать , как создается список. Знать методы преобразования кортежей.
	Лабораторная работа	Списки. Приемы создания списков. Методы и функции работы со списками. Вложенные списки (матрицы). Кортежи. Преобразование кортежей в списки и обратные преобразования. Решение задач с использованием списков и кортежей	Уметь создавать/генерировать список. Уметь использовать методы и функции для работы со списками при решении задач. Уметь создавать вложенные списки (матрицы) и решать задачи с их использованием. Уметь создавать/генерировать кортеж. Уметь использовать методы и функции для работы с кортежами при решении задач.
Тема 7. Словари	Интерактивная лекция	Словари. Создание словаря. Операции над словарями. Этапы изучения темы	Знать понятие «словарь» как тип данных. Знать основные методы и алгоритмы работы со словарями.
	Лабораторная работа	Методы создания и обработки словарей. Решение задач.	Уметь создавать словари. Знать и уметь использовать основные функции и методы работы со словарями при решении задач..

Тема 8. Файлы	<i>Интерактивная лекция</i>	Файлы. Создание файла, чтение из файла, запись в файл. Место темы в курсе программирования.	Знать характеристики текстового файла. Знать синтаксис операций чтения/записи
	<i>Лабораторная работа</i>	Как открыть файл для чтения. Как читать данные из файла построчно. Как считать файл целиком. Как открыть файл для записи. Работа с файлами из Интернета.	Уметь открывать файл для чтения. Уметь читать данные из файла построчно или целиком Уметь читать файлы из Интернета..
Тема 9. Решение задач ЕГЭ по программированию	<i>Практическая работа</i>	Разбор заданий из вариантов ЕГЭ, связанных с программированием.	Уметь решать задачи из вариантов ЕГЭ по программированию.
	Итоговая аттестация	Каждому слушателю предоставляется вариант задач ЕГЭ по программированию. Оцениваются результаты решения задач.	

2.3.2. Рабочая программа: очно-заочная форма обучения

дополнительной профессиональной программы повышения квалификации

ПОДГОТОВКА К ЕГЭ ПО ИНФОРМАТИКЕ: МЕТОДИКА ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ: ВАРИАТИВНЫЙ МОДУЛЬ: ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON (40 час.)

№ п/п	Виды учебных занятий, учебных работ	Содержание	Знания/умения
Тема 1. Особенности изучения программирования на языке Python	<i>Вводная лекция</i>	Особенности языка программирования Python. Версии языка Python. Состав среды программирования. Организация рабочего пространства обучающегося Особенности обучения языку программирования Python..	Знать базовые характеристики и особенности языка программирования Python; отличия версий 2.x и 3.x. Знать , где взять среду программирования Python. Знать средства и методы обучения программирования на языке Python, Уметь запускать среду IDLE.
	<i>Практическое занятие</i>	Основы работы с интерпретатором Python. Арифметические операции. Простейшие типы данных. Технология создания программы в Python. Функции ввода и	Знать арифметические операции. Знать особенности использования операции присваивания в Python. Знать особенности использования функций ввода и вывода. Знать простейшие типы данных, используемых в Python.

		вывода. Основы компьютерной графики.	Уметь использовать интерпретатор для простейших вычисления. Уметь создавать простейшие линейные программы с использованием оператора присваивания, арифметических операций, функций ввода-вывода. Уметь создавать простейшие изображения: круг, эллипс, прямоугольник, линию и многоугольник.
Тема 2. Ветвление	<i>Интерактивная лекция</i>	Основы алгоритмизации: понятие блок-схемы. Полное и неполное ветвление. Особенности решения задачи и оформление программ с конструкцией ветвления в Python. Сложные условия. Методика обучения, прогнозирование типичных ошибок обучающихся.	Знать основные элементы для составления блок-схем. Знать блок-схему для оператора ветвления. Знать назначения операторы AND, OR, NOT. Знать основные приемы преподавания темы/
	<i>Лабораторная работа</i>	Разработка программ с использованием оператора ветвления, содержащего простые и сложные условия, вложенные ветвления, каскадные условия.	Уметь разрабатывать программы, содержащие оператор ветвления. Уметь разрабатывать программы, содержащие вложенные ветвления. Уметь разрабатывать программы, содержащие каскадные и сложные условия.
Тема 3. Реализация циклических алгоритмов	<i>Интерактивная лекция</i>	Циклы на языке Python: цикл for. Циклы с условиями. Типовые алгоритмы. Особенности преподавания темы, вариативность решения задач, подбор задач.	Знать , когда применяется цикл For. Знать , когда используются циклы с условиями. Знать , как проектируются условия для этих циклов.
	<i>Лабораторная работа</i> <i>Самостоятельная работа</i>	Типовые задачи, реализуемые с помощью цикла for. Типовые задачи, реализуемые с помощью цикла while. Типовые алгоритмы: вычисление суммы, определение экстремума, вычисление суммы ряда.	Уметь решать задачи с использованием цикла For. Уметь реализовывать типовые алгоритмы: вычисление суммы и экстремума. Уметь проектировать условия для цикла For. Уметь решать задачи, в которых используются циклы с условием. Уметь проектировать условия для этих циклов. Уметь реализовывать типовые

			алгоритмы: вычисление суммы ряда с заданной точностью.
Тема 4. Функции	<i>Самостоятельная работа</i>	Функции. Рекурсивные алгоритмы на языке Python. Локальные и глобальные переменные. Датчик случайных чисел. Методические приемы изучения темы	Знать , как оформляется функция в Python. Уметь выделять часть алгоритма, который целесообразно оформлять в виде функции. Понимать , как работает рекурсивный алгоритм. Уметь решать задачи, в которых используются рекурсивный алгоритм. Понимать , как работает датчик случайных чисел. Уметь использовать датчик случайных чисел для решения задач.
Тема 5. Алгоритмы работы со строками	<i>Самостоятельная работа</i>	Строковый тип данных. Срезы. Методы работы со строками. Методические аспекты преподавания	Знать , как задается строковая константа в Python. Знать , как делать срезы в Python. Знать методические подходы изучения темы.
	<i>Самостоятельная работа</i>	Как задать строковые константы. Срезы. Базовые функции: конкатенация, дублирование, определение длины строки. Типовые методы и алгоритмы: замена букв/фрагментов в строке; методы поиска, разбиения строк,	Уметь задавать строковые константы. Уметь делать срезы строк. Уметь использовать базовые функции работы со строками для решения задач. Уметь решать задачи с использованием типовых методов и алгоритмов: замена букв/фрагментов в строке; методы поиска, разбиения строк,
Тема 6. Особенности и изучения работы со списками и кортежами	<i>Самостоятельная работа</i>	Операции со списками. Генерация списков. Срезы. Методы сортировки списков. Вложенные списки. Матрицы. Кортежи.	Знать , что такое список. Знать , как создается список. Знать , как делать срезы списков. Знать основные методы и функции, используемые для работы со списками. Знать , что такое кортеж. Знать , как создается кортеж. Знать , как создается список. Знать методы преобразования кортежей.
	<i>Самостоятельная работа</i>	Списки. Приемы создания списков. Методы и функции работы со списками. Вложенные списки (матрицы). Кортежи. Преобразование	Уметь создавать/генерировать список. Уметь использовать методы и функции для работы со списками при решении задач. Уметь создавать вложенные

		кортежей в списки и обратные преобразования. Решение задач с использованием списков и кортежей	списки (матрицы) и решать задачи с их использованием. Уметь создавать/генерировать кортеж. Уметь использовать методы и функции для работы с кортежами при решении задач.
Тема 7. Словари	<i>Самостоятельная работа</i>	Словари. Создание словаря. Операции над словарями. Этапы изучения темы	Знать понятие «словарь» как тип данных. Знать основные методы и алгоритмы работы со словарями.
	<i>Самостоятельная работа</i>	Методы создания и обработки словарей. Решение задач.	Уметь создавать словари. Знать и уметь использовать основные функции и методы работы со словарями при решении задач..
Тема 8. Файлы	<i>Самостоятельная работа</i>	Файлы. Создание файла, чтение из файла, запись в файл. Место темы в курсе программирования.	Знать характеристики текстового файла. Знать синтаксис операций чтения/записи
	<i>Самостоятельная работа</i>	Как открыть файл для чтения. Как читать данные из файла построчно. Как считать файл целиком. Как открыть файл для записи. Работа с файлами из Интернета.	Уметь открывать файл для чтения. Уметь читать данные из файла построчно или целиком Уметь читать файлы из Интернета..
Тема 9. Решение задач ЕГЭ по программированию	<i>Практическая работа</i>	Разбор заданий из вариантов ЕГЭ, связанных с программированием.	Уметь решать задачи из вариантов ЕГЭ по программированию.
	Итоговая аттестация	Каждому слушателю предоставляется вариант задач ЕГЭ по программированию. Оцениваются результаты решения задач.	

2.3.3. Рабочая программа: заочная форма обучения с использованием ДОТ

дополнительной профессиональной программы повышения квалификации

ПОДГОТОВКА К ЕГЭ ПО ИНФОРМАТИКЕ: МЕТОДИКА ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ: ВАРИАТИВНЫЙ МОДУЛЬ: ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON (40 час.)

№ п/п	Виды учебных занятий, учебных работ	Содержание	Знания/умения
Тема 1. Особенности изучения программирования на языке Python	<i>Видео-лекция</i>	Особенности языка программирования Python. Версии языка Python. Состав среды программирования. Организация рабочего пространства обучающегося. Особенности обучения языку программирования Python..	Знать базовые характеристики и особенности языка программирования Python; отличия версий 2.x и 3.x. Знать , где взять среду программирования Python. Знать средства и методы обучения программирования на языке Python, Уметь запускать среду IDLE.
	<i>Самостоятельная работа</i>	Основы работы с интерпретатором Python. Арифметические операции. Простейшие типы данных. Технология создания программы в Python. Функции ввода и вывода. Основы компьютерной графики.	Знать арифметические операции. Знать особенности использования операции присваивания в Python. Знать особенности использования функций ввода и вывода. Знать простейшие типы данных, используемых в Python. Уметь использовать интерпретатор для простейших вычисления. Уметь создавать простейшие линейные программы с использованием оператора присваивания, арифметических операций, функций ввода-вывода. Уметь создавать простейшие изображения: круг, эллипс, прямоугольник, линию и многоугольник.
Тема 2. Ветвление	<i>Самостоятельная работа</i>	Основы алгоритмизации: понятие блок-схемы. Полное и неполное ветвление. Особенности решения задачи и оформление программ с конструкцией ветвления в	Знать основные элементы для составления блок-схем. Знать блок-схему для оператора ветвления. Знать назначения операторы AND, OR, NOT. Знать основные приемы

		Python. Сложные условия. Методика обучения, прогнозирование типичных ошибок обучающихся.	преподавания темы/
	<i>Самостоятельная работа</i>	Разработка программ с использованием оператора ветвления, содержащего простые и сложные условия, вложенные ветвления, каскадные условия.	Уметь разрабатывать программы, содержащие оператор ветвления. Уметь разрабатывать программы, содержащие вложенные ветвления. Уметь разрабатывать программы, содержащие каскадные и сложные условия.
Тема 3. Реализация циклических алгоритмов	<i>Самостоятельная работа</i>	Циклы на языке Python: цикл for. Циклы с условиями. Типовые алгоритмы. Особенности преподавания темы, вариативность решения задач, подбор задач.	Знать , когда применяется цикл For. Знать , когда используются циклы с условиями. Знать , как проектируются условия для этих циклов.
	<i>Самостоятельная работа</i>	Типовые задачи, реализуемые с помощью цикла for. Типовые задачи, реализуемые с помощью цикла while. Типовые алгоритмы: вычисление суммы, определение экстремума, вычисление суммы ряда.	Уметь решать задачи с использованием цикла For. Уметь реализовывать типовые алгоритмы: вычисление суммы и экстремума. Уметь проектировать условия для цикла For. Уметь решать задачи, в которых используются циклы с условием. Уметь проектировать условия для этих циклов. Уметь реализовывать типовые алгоритмы: вычисление суммы ряда с заданной точностью.
Тема 4. Функции	<i>Самостоятельная работа</i>	Функции. Рекурсивные алгоритмы на языке Python. Локальные и глобальные переменные. Датчик случайных чисел. Методические приемы изучения темы	Знать , как оформляется функция в Python. Уметь выделять часть алгоритма, который целесообразно оформлять в виде функции. Понимать , как работает рекурсивный алгоритм. Уметь решать задачи, в которых используются рекурсивный алгоритм. Понимать , как работает датчик случайных чисел. Уметь использовать датчик случайных чисел для решения задач.
Тема 5. Алгоритмы	<i>Самостоятельная работа</i>	Строковый тип данных. Срезы. Методы работы со	Знать , как задается строковая константа в Python.

работы со строками		строками. Методические аспекты преподавания	Знать , как делать срезы в Python. Знать методические. подходы изучения темы.
	<i>Самостоятельная работа</i>	Как задать строковые константы. Срезы. Базовые функции: конкатенация, дублирование, определение длины строки. Типовые методы и алгоритмы: замена букв/фрагментов в строке; методы поиска, разбиения строк,	Уметь задавать строковые константы. Уметь делать срезы строк. Уметь использовать базовые функции работы со строками для решения задач. Уметь решать задачи с использованием типовых методов и алгоритмов: замена букв/фрагментов в строке; методы поиска, разбиения строк,
Тема 6. Особенности и изучения работы со списками и кортежами	<i>Самостоятельная работа</i>	Операции со списками. Генерация списков. Срезы. Методы сортировки списков. Вложенные списки. Матрицы. Кортежи.	Знать , что такое список. Знать , как создается список. Знать , как делать срезы списков. Знать основные методы и функции, используемые для работы со списками. Знать , что такое кортеж. Знать , как создается кортеж. Знать , как создается список. Знать методы преобразования кортежей.
	<i>Самостоятельная работа</i>	Списки. Приемы создания списков. Методы и функции работы со списками. Вложенные списки (матрицы). Кортежи. Преобразование кортежей в списки и обратные преобразования. Решение задач с использованием списков и кортежей	Уметь создавать/генерировать список. Уметь использовать методы и функции для работы со списками при решении задач. Уметь создавать вложенные списки (матрицы) и решать задачи с их использованием. Уметь создавать/генерировать кортеж. Уметь использовать методы и функции для работы с кортежами при решении задач.
Тема 7. Словари	<i>Самостоятельная работа</i>	Словари. Создание словаря. Операции над словарями. Этапы изучения темы	Знать понятие «словарь» как тип данных. Знать основные методы и алгоритмы работы со словарями.
	<i>Самостоятельная работа</i>	Методы создания и обработки словарей. Решение задач.	Уметь создавать словари. Знать и уметь использовать основные функции и методы работы со словарями при решении задач..
Тема 8.	<i>Самостоятельная</i>	Файлы. Создание файла,	Знать характеристики текстового

Файлы	<i>работа</i>	чтение из файла, запись в файл. Место темы в курсе программирования.	файлы. Знать синтаксис операций чтения/записи
	<i>Самостоятельная работа</i>	Как открыть файл для чтения. Как читать данные из файла построчно. Как считать файл целиком. Как открыть файл для записи. Работа с файлами из Интернета.	Уметь открывать файл для чтения. Уметь читать данные из файла построчно или целиком Уметь составлять программы, которые считывают файлы из Интернета.
Тема 9. Решение задач ЕГЭ по программированию	<i>Вебинар (заочный семинар)</i>	Разбор заданий из вариантов ЕГЭ, связанных с программированием.	Уметь решать задачи из вариантов ЕГЭ по программированию.
	Итоговая аттестация	Каждому слушателю предоставляется вариант задач ЕГЭ по программированию. Оцениваются результаты решения задач.	

Раздел 3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

дополнительной профессиональной программы повышения квалификации

ПОДГОТОВКА К ЕГЭ ПО ИНФОРМАТИКЕ:

МЕТОДИКА ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ:

ВАРИАТИВНЫЙ МОДУЛЬ:

ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON

(40 час.)

Форма обучения	Общая продолжительность ДПП (календарных дней)	Режим занятий (кол-во час.) в день	Количество часов ДПП	Лекции (кол-во час.)	Практ. занятия (кол-во час.)	Сам. раб. (кол-во час.)	Промежуточная аттестация (кол-во час., вид ПА)	Итоговая аттестация (кол-во час., вид ИА)
очная	5	8	40	7	33	0	0	4
очно-заочная с ДОТ	23	8 в очной части, 1-2 в ДОТ	40	2	10	24	Контрольные работы: 2	4
заочная с ДОТ	30	1-2 часа в день	40	0	0	36	Контрольные работы: 2	4