

СПЕЦИФИКАЦИЯ

диагностической работы по информатике для 10-х классов (профиль)

1. Назначение диагностической работы

Диагностическая работа проводится с целью определения:

- уровня овладения навыками алгоритмического мышления,
- уровня сформированности самостоятельного и творческого подхода к решению задач,
- уровня навыков работы с современным ПО (среды программирования PyCharm и IDLE), учащимися профильных 10-х классов общеобразовательных организаций.

2. Документы, определяющие содержание и параметры диагностической работы

Содержание и основные характеристики проверочных материалов определяются на основе следующих документов:

- Федерального государственного образовательного стандарта среднего общего образования (Приказ Министерства образования и науки Российской Федерации от 17 мая 2012 г. № 413 «Об утверждении федерального государственного образовательного стандарта среднего общего образования»; приказ Министерства просвещения Российской Федерации от 12 августа 2022 г. N 732 «О внесении изменений в федеральный государственный образовательный стандарт среднего общего образования, утвержденный приказом Министерства образования и науки Российской Федерации от 17 мая 2012 г. № 413»);
- Образовательной программы среднего общего образования ГАОУ СО «Гимназия № 1»;
- Федеральной рабочей программы по учебному предмету «Информатика»;
- Приказа Министерства просвещения Российской Федерации от 14.09.2021 г. № 03-1510 «Об организации работы по повышению функциональной грамотности»;
- Универсальные кодификаторы распределенных по классам проверяемых элементов содержания и требований к результатам освоения основной образовательной программы основного общего образования (одобрены решением федерального учебно-методического объединения по общему образованию (протокол от 12.04.2021 г. №1/21)).3.

Условия проведения диагностической работы

Работа проводится полностью в компьютерной форме. Пользоваться собственными записями, бумажными или электронными книгами запрещено.

На компьютере должна быть запущена любая среда программирования (ученики может выбрать на свое усмотрение: либо PyCharm, либо IDLE), на столах — только ручка и бумага. Решение должно быть сохранено учеником в

указанную ему на экзамене папку, записи в черновиках не учитываются и не оцениваются.

4. Время выполнения диагностической работы.

На выполнение диагностической работы отводится 60 минут.

5. Содержание и структура диагностической работы

Диагностическая работа представляет собой список примерных задач на основные алгоритмы, которые должен знать ученик профильного класса.

Все задачи будут разбиты на билеты по 3 задачи в каждом.

Каждую из предложенных задач можно решить:

1. с использованием функций и методов языка Python (базовый уровень),
2. не пользуясь ими, а реализуя самостоятельно необходимые алгоритмы: подсчета, поиска, сортировки,... (повышенный уровень),
3. написать собственную функцию, которую использовать при решении основной задачи (продвинутый уровень).

На уроках информатики формируем особый вид грамотности – информационную и функциональную, которая предполагает:
- умение искать информацию, критически её оценивать, выбирать нужную, использовать её и создавать новую (любая из задач списка)

В работу включены задания №1, №2, №3, №4, №5, №7 проверяющие **функциональную** грамотность учащихся.

6. Порядок оценивания выполнения диагностической работы

Оценка за экзамен зависит от количества решенных задач и от того, какое из трех условий, перечисленных выше, было выполнено.

Если был выбран 1 уровень и решены все 3 задачи — ставится отметка «3», если решено менее 3 задач - «2»

Таблица 1

Выбранный уровень решения	Количество баллов
1) базовый	2 балла – верно решены обе задачи; 1 балл – решена только 1 задача. 0 баллов – ни 1 задача не решена. При решении задачи использовались встроенные функции языка Питон.
2) повышенный	4 балла – верно решены обе задачи, встроенные функции и методы не использовались; 2 балла – решена 1 задача, встроенные функции и методы не использовались;

	<p>Или решены все 2 задачи, но в решении 1 из задач (№1 или №2) использовались встроенные функции Python. 0 баллов — не выполнены требования на 4 или 2 балла.</p>
<p>3) продвинутый</p>	<p>6 баллов – верно решены обе задачи, встроенные функции и методы не использовались; была написана собственная функция, которая использовалась при решении 3 балла – решена 1 задача, встроенные функции и методы не использовались; была написана собственная функция, которая использовалась при решении. 0 баллов — не выполнены требования на 6 или 3 балла.</p>

Список примерных вопросов для составления билетов.

1. Найдите минимальную и максимальную стоимость товара из четырех вариантов цен без использования списков и циклов.
2. Квадратное уравнение задано своими коэффициентами. Определите количество корней уравнения и их значения.
3. Дано число в 10-й системе счисления. Найдите сумму его четных цифр в 3-й системе счисления.
4. Дано число в 10-й системе счисления. Переведите его в систему счисления с основанием $k \leq 10$.
5. Дано число в системе счисления с основанием k . Переведите его в 10-ю систему счисления.
6. Найдите сумму всех отрицательных элементов списка, стоящих на четных местах.
7. Напишите программу, реализующую решето Эратосфена.
8. Найдите наименьший простой делитель числа.
9. Заполните список по следующему правилу: значение элемента списка равно факториалу удвоенного значения его индекса. Количество элементов списка вводится с клавиатуры.
10. Заполните список случайными числами из отрезка $[a, b]$. Найдите индексы всех вторых максимумов за один проход списка.
11. Создайте список из n элементов в диапазоне значений от -15 до 14 включительно. Определите количество элементов по модулю больших, чем минимальный четный.
12. Заполните список случайными положительными и отрицательными целыми числами и выведите его на экран. Удалите из списка все отрицательные элементы и снова выведите.
13. Считайте элементы списка, введенные через пробел. Запросите еще одно значение и его позицию в списке, после которого нужно вставить и количество. Вставьте указанное число в заданную позицию, в указанном количестве, подвинув элементы после него.
14. Отсортируйте нечетные элементы списка, заполненного случайными числами, по возрастанию. Для сортировки используйте метод "пузырька". Выведите на экран список в исходном и отсортированном виде.
15. Дан список, заполненный случайными числами. Отсортируйте элементы списка после последнего отрицательного элемента по убыванию. Для сортировки используйте метод выбора. Выведите на экран список в исходном и отсортированном виде.
16. Дана строка. Подсчитайте количество появления каждого ее символа.
17. Дано число N и далее N строк. В последней строке дано еще одно слово. Выведите через пробел все строки, в которых не оказалось искомого слова в нижнем регистре.
18. Дано число N и далее N строк. Отсортируйте введенные строки по возрастанию в алфавитном порядке (каждая следующая строка должна быть больше предыдущей с точки зрения оператора сравнения «>» в Питоне; в частности, все строчные буквы идут после всех заглавных). Встроенные сортировки не использовать.
19. Дано число N и далее N строк. Отсортируйте введенные строки по возрастанию их длины, строки одинаковой длины при этом должны быть отсортированы по возрастанию в алфавитном порядке. Встроенные сортировки не использовать.

Примеры билетов.

Билет 1.

1. Дано число в 10-й системе счисления. Найдите произведение его нечетных цифр в системе счисления с основанием k ($k \leq 10$).
2. Считайте элементы списка, введенные через пробел. Запросите еще одно значение и его позицию в списке, после которого нужно вставить это значение и количество. Вставьте указанное число в заданную позицию, в указанном количестве, подвинув элементы после него. Встроенные функции не использовать.

Билет 2.

1. Напишите программу, реализующую решето Эратосфена. Количество элементов списка вводится с клавиатуры.
2. Отсортируйте отрицательные элементы списка, заполненного случайными числами, по убыванию. Для сортировки используйте метод "пузырька". Выведите на экран список в исходном и отсортированном виде.