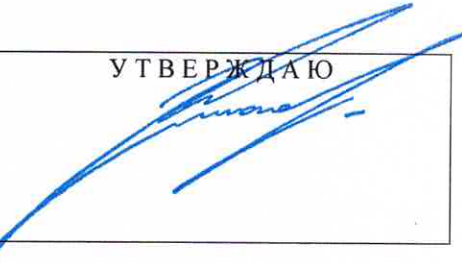
 Минобрнауки России МИРЭА – Российский технологический университет	Вступительное испытание по основам алгоритмизации и программирования 2024 год  Вариант № 1	УТВЕРЖДАЮ 
---	---	---

1. Для перечисленных в единственной строке вещественных чисел  $A_1, A_2, A_3, A_4, A_5$ , разделенных пробелом, необходимо найти результат вычисления следующей функции. В процессе вычислений результат **каждой** операции, изменяющей значение числа, необходимо округлять до 5 знаков после запятой. Операции с одинаковым приоритетом выполнять слева направо. Итоговое значение необходимо округлить до 2 знаков после запятой.

$$\ln A_1 - \sqrt{\frac{A_5}{A_3}} * A_5^{\sqrt{A_5}} + \frac{\sin(A_2)}{\tan(A_2 - \frac{A_4}{A_3})}$$

$A_2$  и  $A_4$  указаны в радианах.

Входные данные – в файле **in-1-01.txt**.

Ответ: \_\_\_\_\_

2. Для перечисленных в единственной строке вещественных чисел  $A_1, A_2, A_3, A_4, A_5$ , разделенных пробелом, необходимо найти результат вычисления следующей функции. В процессе вычислений результат **каждой** операции, изменяющей значение числа, необходимо округлять до 5 знаков после запятой. Операции с одинаковым приоритетом выполнять слева направо. Итоговое значение необходимо округлить до 2 знаков после запятой.

$$A_2^{|\cos(A_5) * 2|} * \sqrt{\left| \frac{\sqrt{A_4}}{A_3} \right|} - \frac{\tan(A_1 + A_5)}{A_2}$$

$A_1$  указан в радианах.

$A_5$  указан в градусах.

Входные данные – в файле **in-1-02.txt**.

Ответ: \_\_\_\_\_

3. Для перечисленных в единственной строке вещественных чисел  $A_1, A_2, A_3, A_4, A_5$ , разделенных пробелом, необходимо найти результат вычисления следующей функции. В процессе вычислений результат **каждой** операции, изменяющей значение числа, необходимо округлять до 5 знаков после запятой. Итоговое значение необходимо округлить до 2 знаков после запятой.



$$\frac{A_1^{A_5}}{\sqrt{B_2}} * \sin A_2 + \ln B_1$$

$$B_1 = \begin{cases} A_1 * A_5 & \text{при } A_4 < 5.7 \\ A_3 & \text{при } A_4 \geq 5.7 \end{cases}$$

$$B_2 = \begin{cases} A_5^{A_1} & \text{при } B_1 \geq 15 \\ \sqrt{A_4} & \text{при } B_1 < 15 \end{cases}$$

$A_2$  указан в градусах.

Входные данные – в файле **in-1-03.txt**.

 <p>Минобрнауки России МИРЭА – Российский технологический университет</p>	<p>Вступительное испытание по основам алгоритмизации и программирования 2024 год</p> <p>Вариант № 1</p>	<p>УТВЕРЖДАЮ</p> 
---	---	--

Ответ: \_\_\_\_\_

4. Для перечисленных в единственной строке вещественных чисел  $A_1, A_2, A_3, A_4, A_5$ , разделенных пробелом, необходимо найти результат вычисления следующей функции. В процессе вычислений результат **каждой** операции, изменяющей значение числа, необходимо округлять до 5 знаков после запятой. Операции с одинаковым приоритетом выполнять слева направо. Итоговое значение необходимо округлить до 2 знаков после запятой.

$$-\frac{\cos(A_5)}{\ln A_2} * \sqrt{\frac{A_4}{B_1}} * \log_{10} B_1 + B_2$$

$$B_1 = \begin{cases} \frac{A_3}{A_2} & \text{при } A_1 < 86.4 \\ A_4 & \text{при } A_1 \geq 86.4 \end{cases}$$

$$B_2 = \begin{cases} \sqrt{A_1} & \text{при } B_1 \geq 178 \\ B_1^4 & \text{при } B_1 < 178 \end{cases}$$

$A_5$  указан в радианах.

Входные данные – в файле **in-1-04.txt**.

Ответ: \_\_\_\_\_

5. В первой строке вводятся два положительных целых числа  $N$ , не превышающее 10000, и  $M$ , не превышающее 200. Далее в  $N$  строках вводятся целые числа, по модулю не превышающие 1000000. Необходимо найти количество чисел, кратных  $M$ .

Входные данные – в файле **in-1-05.txt**.

Ответ: \_\_\_\_\_

6. В первой строке вводятся два положительных целых числа  $N$ , не превышающее 10000, и  $M$ , не превышающее 10. Далее в  $N$  строках вводятся  $N$  целых чисел, по модулю не превышающих 1000000. Необходимо найти количество чисел, больше  $M$ . При этом необходимо анализировать не все числа, а пропуская на каждом шаге вдвое больше чисел чем на прошлом шаге, начиная с 1. Таким образом необходимо проверять числа со следующим порядковым номером: 1, (пропуск одного), 3, (пропуск двух), 6, (пропуск четырех), 11, (пропуск 8), 20, и так далее.

Входные данные – в файле **in-1-06.txt**.


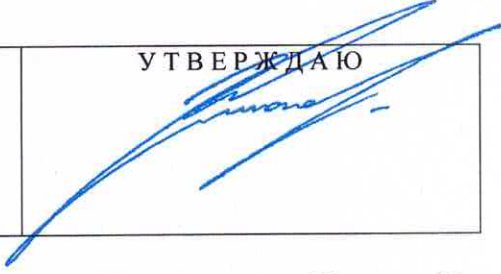
Ответ: \_\_\_\_\_

7. Для набора чисел необходимо определить, возможно ли при помощи операций «\*8» и «\*3» достигнуть этих чисел, начиная с единицы. Например, число 72 получить можно (1,\*8,\*3,\*3), а число 11 – никак нельзя. Если число получить можно – ответить «Да», в противном случае ответить «Нет». Результаты записать через пробел для каждого входного соответственно.

Входные данные – в файле **in-1-07.txt**.

Ответ: \_\_\_\_\_



 <p>Минобрнауки России МИРЭА – Российский технологический университет</p>	<p>Вступительное испытание по основам алгоритмизации и программирования 2024 год</p> <p>Вариант № 1</p>	<p>УТВЕРЖДАЮ</p> 
---	---	--

8. Для набора чисел необходимо определить, возможно ли при помощи операций «\*7» и «+13» достигнуть этих чисел, начиная с единицы. При этом максимально допустимое количество операций – 80. Например, число 3220 получить можно (1,\*7,\*7,+13,\*7,+13,+13,\*7), а число 9 – никак нельзя. Если число получить можно – ответить «Да», в противном случае ответить «Нет». Результаты записать через пробел для каждого входного соответственно.

Входные данные – в файле **in-1-08.txt**.

Ответ: \_\_\_\_\_

9. В первой строке подается положительное целое четное число  $N$  не превышающее 100000, определяющее количество данных. Далее в  $N$  строках подаются целые числа, не превышающие по модулю 1000000. Необходимо найти пару чисел отличных от 0, среднее арифметическое которых кратно 12. Вывести их через пробел. Первое число пары берется из первой половины данных, второе – из второй, при этом числа должны быть равноудалены от середины. То есть парами могут являться первое число и последнее, второе и предпоследнее, и так далее. При наличии нескольких пар вывести самую удаленную от центра. При отсутствии пар, удовлетворяющих условию ответить «NO».

Входные данные – в файле **in-1-09.txt**.

Ответ: \_\_\_\_\_

10. В первой строке подается положительное целое четное число  $N$  не превышающее 100000, определяющее количество данных. Далее в  $N$  строках подаются целые числа, не превышающие по модулю 1000000. Необходимо найти ближайшую к центру пару чисел отличных от 0, произведение которых кратно 17, и вывести их через пробел. Первое число пары берется из первой половины данных, второе – из второй, при этом числа должны быть равноудалены от середины. То есть парами могут являться первое число и последнее, второе и предпоследнее, и так далее. При наличии нескольких пар вывести ближайшую к центру. При отсутствии пар, удовлетворяющих условию ответить «NO».

Входные данные – в файле **in-1-10.txt**.

Ответ: \_\_\_\_\_



11. В первой строке подается положительное целое четное число  $N$  не превышающее 100000, определяющее количество данных. Далее в  $N$  строках подаются строки, содержащие сначала имя, потом фамилию, разделенные символом «~». Необходимо отсортировать строки по фамилии в алфавитном порядке, после чего вывести количества строк в первых десяти группах фамилий. Вывести их через пробел.

Входные данные – в файле **in-1-11.txt**.

Ответ: \_\_\_\_\_

12. В первой строке подается положительное целое четное число  $N$ , определяющее количество данных. Далее в  $N$  строках подаются строки, содержащие номера международных телефонов в разнообразных форматах. Необходимо вывести количество номеров, длина которых равна **минимальной** среди предоставленных. Номер анализируется полностью, с учетом символьных знаков.



 <p>Минобрнауки России МИРЭА – Российский технологический университет</p>	<p>Вступительное испытание по основам алгоритмизации и программирования 2024 год</p> <p>Вариант № 1</p>	<p>УТВЕРЖДАЮ</p> 
---	---	--

Входные данные – в файле **in-1-12.txt**.

Ответ: \_\_\_\_\_

13. В первой строке подается два положительных целых числа **N** и **M** не превышающих 1000, определяющих размер двумерной матрицы. Далее в **N** строках подается по **M** положительных целых чисел, разделенных пробелами, не превышающих по модулю 1000000. Необходимо вывести количество пар чисел, расположенных вертикально рядом, сумма которых кратна 13. Одно число может одновременно относиться к двум парам.

Входные данные – в файле **in-1-13.txt**.

Ответ: \_\_\_\_\_

14. В первой строке подается положительное целое число **N**, не превышающее 1000, определяющее размер квадратной матрицы. Далее в **N** строках подается по **N** целых чисел, разделенных пробелами, не превышающих по модулю 1000000. Необходимо найти среднее арифметическое элементов, расположенных на диагоналях матрицы. Результат округлить до 2 знаков после запятой.

Входные данные – в файле **in-1-14.txt**.

Ответ: \_\_\_\_\_

15. Предоставляется файл **in-1-15.txt** с положительным числом строк, содержащих латинские символы и цифры. Необходимо найти строки, заканчивающиеся последовательностью «RTU». Результатом работы программы должна быть строка, где через пробел указываются количества букв R, T и U соответственно во всех найденных строках. Регистр букв имеет значение.

Входные данные – в файле **in-1-15.txt**.

Ответ: \_\_\_\_\_

16. Предоставляется файл **in-1-16.txt** с положительным числом строк, содержащих латинские символы и цифры. Необходимо вывести количества нечетных цифр меньше 7, встречающихся в файле, в формате «цифра-количество,цифра-количество». Цифры перечислять в порядке возрастания.

Входные данные – в файле **in-1-16.txt**.

Ответ: \_\_\_\_\_

17. Представлен код, описывающий класс «Показания температуры», и создающий его экземпляр, который может сохранять собранные целочисленные значения температуры в тысячных градуса Цельсия. Необходимо написать код, который добавит 3 измерения по 4 единицы и одно измерение в 17 единиц в созданный объект. В первой строке необходимо указать язык программирования, на котором вы предоставляете ответ.

C#	C++
<pre>class TemperatureReadings {     private readonly List&lt;int&gt; readings = new List&lt;int&gt;();     public void AddReading(int reading) { readings.Add(reading); } }</pre>	<pre>#include &lt;list&gt; class TemperatureReadings{     private:         std::list&lt;int&gt; readings = {};     public:         void AddReading(int reading) {             readings.push_back(reading);         } }</pre>



<pre>public class Program {     public static void Main(string[] args)     {         TemperatureReadings tReadings = new TemperatureReadings();         // Место для кода     } }</pre>	<pre>}; int main() {     TemperatureReadings* tReadings = new TemperatureReadings();     // Место для кода     return 0; }</pre>
<b>Java</b>	<b>Python</b>
<pre>import java.util.ArrayList; public class Program {     public static void main(String args[]) {         TemperatureReadings tReadings = new TemperatureReadings();         // Место для кода     } }  class TemperatureReadings {     private final ArrayList&lt;Integer&gt; readings = new ArrayList&lt;Integer&gt;();     public void addReading(int reading)     {         readings.add(reading);     } }</pre>	<pre>class TemperatureReadings:     def __init__(self) -&gt; None:         self.readings = []      def addReading(self, reading) -&gt; None:         self.readings.append(reading)  t_readings = TemperatureReadings() # Место для кода</pre>
<b>JavaScript</b>	
<pre>class TemperatureReadings {     #readings;     constructor() {         this.#readings = [];     }     addReading(reading) {         this.#readings.push(reading);     } }  const tReadings = new TemperatureReadings(); // Место для кода</pre>	

Ответ:

18. Представлен код, описывающий класс «Показания», и создающий его экземпляр, который может хранить показания различного вида. Необходимо написать код, который добавит 2 измерения температуры, в 3 и 16 единиц, и одно значение влажности в 40 единиц в созданный объект. В первой строке необходимо указать язык программирования, на котором вы предоставляете ответ.

C#

C++





```
class Meter
{
    private readonly List<Measurement>
measurements = new List<Measurement>();
    public void AddMeasurement(Measurement
measurement) {
measurements.Add(measurement); }
}
abstract class Measurement { }
class Temperature(double value) :
Measurement{ }
class AirHumidity(int value) : Measurement{
}

public class Program
{
    public static void Main(string[] args)
    {
        Meter meter = new Meter();
        // Место для кода
    }
}
```

```
#include <list>
class Measurement {};
class Meter {
private:
    std::list<Measurement*>
measurements = {};
public:
    void
AddMeasurement(Measurement*
measurement) {
measurements.push_back(measurement);
}
};
class Temperature : public Measurement
{
private:
    double value;
public:
    Temperature(double value) {
this->value = value;
};
class AirHumidity : public Measurement
{
private:
    int value;
public:
    AirHumidity(int value) { this-
>value = value;
};
int main()
{
    Meter* meter = new Meter();
    // Место для кода
    return 0;
}
```

**Java**

```
import java.util.ArrayList;
public class Program {
    public static void main(String
args[]) {
        Meter meter = new Meter();
        // Место для кода
    }
}

class Meter
{
    private final ArrayList<Measurement>
measurements = new
ArrayList<Measurement>();
    public void
addMeasurement(Measurement measurement)
    {
        measurements.add(measurement);
    }
}
```

**Python**

```
from typing import List
class Measurement: pass
class Temperature(Measurement):
    def __init__(self, value: float) ->
None:
        self.value = value
class AirHumidity(Measurement):
    def __init__(self, value: int) ->
None:
        self.value = value
class Meter:
    measurements: List[Measurement]
    def __init__(self) -> None:
        self.measurements = []
    def addMeasurement(self, measurement:
Measurement) -> None:
        self.measurements.append(measurement)
```



<pre>    } } abstract class Measurement { } class Temperature extends Measurement {     private double value;     public Temperature(double value){ this.value = value; } } class AirHumidity extends Measurement {     private int value;     public AirHumidity(int value){ this.value = value; } }</pre>	<pre>meter = Meter() # Место для кода</pre>
<p style="text-align: center;"><b>JavaScript</b></p> <pre>class Measurement { } class Temperature extends Measurement {     #count;     constructor(count) {         super();         this.count = count;     } } class AirHumidity extends Measurement {     #count;     constructor(count) {         super();         this.count = count;     } } class Meter {     #measurements;     constructor() {         this.#measurements = [];     }     addMeasurement(measurement) { this.#measurements.push(measurement);     } } const meter = new Meter(); // Место для кода</pre>	

Ответ:



19. Представлен код, где описан код сущности «Двигатель» (Engine), и его наследника – «Турбо двигатель» (TurboEngine). Необходимо переопределить поведение метода «Получить мощность» (getpower), таким образом, чтобы возвращаемая мощность была равна 130% от изначальной мощности. В первой строке необходимо указать язык программирования, на котором вы предоставляете ответ.

C#	C++
<pre>class Engine {     protected double power = 70;     public virtual double GetPower() { return power; } } class TurboEngine : Engine {     // Место для кода }</pre>	<pre>class Engine {     protected:         double power = 70;     public:         virtual double GetPower() { return power; } }; class TurboEngine : public Engine {     public:         // Место для кода };</pre>
Java	Python
<pre>class Engine {     protected double power = 70;     public double getPower(){ return power; } } class TurboEngine extends Engine {     // Место для кода }</pre>	<pre>class Engine():     power = 70     def getPower(self):         return self.power class TurboEngine(Engine):     # Место для кода     pass</pre>
JavaScript	
<pre>class Engine {     _power = 100     getPower() {         return this._power;     } } class TurboEngine extends Engine {     // Место для кода }</pre>	





Минобрнауки России  
МИРЭА – Российский  
технологический университет

Вступительное испытание  
по основам алгоритмизации и  
программирования  
2024 год

Вариант № 1

УТВЕРЖДАЮ

Ответ:

20. Необходимо написать класс “UserName”, содержащий одно приватное строковое поле «name», равное по умолчанию «Максим». Класс должен содержать два публичных метода: «getName()» и «setName(value)». Первый – возвращает значение поля. Второй – заменяет хранимое значение переданным в единственном аргументе именем. Если имя пустое – изменение не должно происходить. В первой строке необходимо указать язык программирования, на котором вы предоставляете ответ.

Ответ:

21. На основании данных из файла **in-1-21.txt** необходимо провести расчет описываемых преобразований над числом 1, и найти итоговое значение. Файл содержит по одному преобразованию на каждой строке. Каждое преобразование состоит из символа действия «-», «+», «\*», и целого числа, не превышающего по модулю  $2^{64}$ , с которым это действие необходимо произвести. Таким образом при следующем содержимом файла

+ 15

- 3

\* 2

Результат будет  $((1 + 15) - 3) * 2 = 26$ . Гарантируется, что при выполнении операций не будет получено число, превышающее по модулю 10000000.

Входные данные – в файле **in-1-21.txt**.

Ответ: \_\_\_\_\_

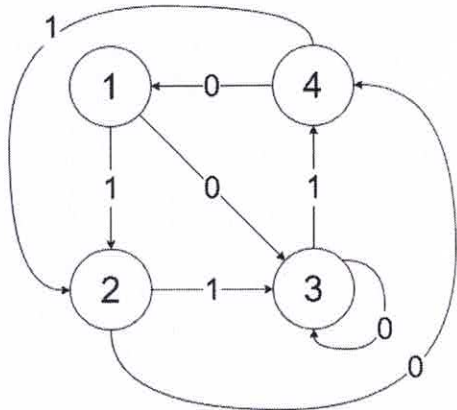
22. На основании данных из файла **in-1-22.txt** необходимо выполнить операции над матрицей. Результатом будет являться сумма её элементов. В первой строке подается три положительных целых

числа **N**, **M**, и **K**, определяющих размер двумерной матрицы и количество операций соответственно. Далее в **N** строках подается по **M** целых чисел, разделенных пробелами, не превышающих по модулю 1000000. После информации о изначальном состоянии матрицы передается **K** строк, содержащие команды для модификации состояния матрицы. Каждая команда состоит из трех чисел. Первые два указывают на порядковые номера строки и колонки изменяемого значения соответственно. Третье – целое число, на которое необходимо заменить найденное значение, не превышает по модулю 1000000. Порядковые номера строк и колонок начинаются с 0. Гарантируется, что при выполнении операций не будет получено число, превышающее по модулю 10000000000.

Входные данные – в файле **in-1-22.txt**.

Ответ: \_\_\_\_\_

23. Игроку дается граф, состоящий из четырех узлов, при этом из каждого узла игрок может перейти в какой-то узел двумя дорогами, обозначенными цифрами 0 и 1. Изначально игрок находится в узле номер 1. Файл **in-1-23.txt** содержит в себе строки, содержащие по одному символу «0» или «1», определяющие последовательность действий. Необходимо найти пять номеров узлов, которыми закончится путь игрока при выполнении всех перемещений.


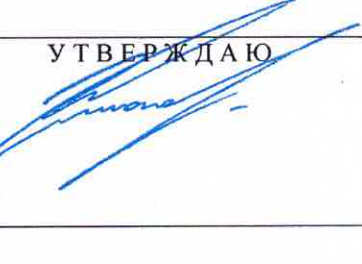


Входные данные – в файле **in-1-23.txt**.  
 Ответ: \_\_\_\_\_

24. В файле **in-1-24.txt** содержится информация о сталактитах и сталагмитах. Наличие породы обозначается точкой, отсутствие пробелом. Все сталагмиты начинаются с нижней строки, все сталактиты с верхней. Гарантируется, что в каждой колонке будет как минимум один пробел (1 единица расстояния), таким образом всегда можно разделить столбец на сталактит и сталагмит. Необходимо найти среднее арифметическое расстояний между парными сталактитами и сталагмитами. Результат округлить до 2 знаков после запятой.

.	.	.	.
.	.	.	.
.	.		
.	.		
	.		
	.	.	
.	.	.	
.	.	.	
.	.	.	.



 <p>Минобрнауки России МИРЭА – Российский технологический университет</p>	<p>Вступительное испытание по основам алгоритмизации и программирования 2024 год</p> <p>Вариант № 1</p>	<p>УТВЕРЖДАЮ</p> 
---	---	---

Входные данные – в файле **in-1-24.txt**.

Ответ: \_\_\_\_\_

25. Рассматривается спутник Земли, движущийся по орбите. Моментальную высоту орбиты возьмем за промежуток координат  $-1000000000$  до  $1000000000$ , где нижняя граница – гарантированное столкновение с планетой, а верхняя – сход с орбиты. Изначально спутник находится в координате 0. Определенные наблюдения за поверхностью можно производить исключительно на определенной высоте, из-за устройства линз на спутнике. Необходимо выяснить, сколько раз спутник будет попадать в этот промежуток высоты для снятия показания. На первой строке подается три целых числа, промежуток высоты **P1** и **P2**, и количество изменений высоты спутника **N**. Далее в **N** строках передаются ненулевые расстояния, на которые должен сместиться спутник. Считается, что спутник попадает в промежуток в случаях, когда его движение включало в себя требуемый промежуток высоты, или он закончил свое перемещение внутри этого промежутка, включая P1 и P2. При этом движение должно начинаться вне целевого промежутка.

Входные данные – в файле **in-1-25.txt**.

Ответ: \_\_\_\_\_

 Белая П.В.