**Примерный вариант**

Экзаменационная работа состоит 5 разделов, включающих 12 заданий. По каждому заданию указано максимальное количество баллов, которое можно получить при успешном его выполнении.

1. Вычислите выражение 21103 – 489 + 510. Ответ дайте в десятичной системе счисления.

**(6 баллов)**

2. Михаил составляет 5-буквенные коды. В кодах разрешается использовать только буквы А, Б, В, Г, Д, при этом код не может начинаться с гласной и не может содержать двух одинаковых букв подряд. Сколько различных кодов может составить Михаил?

**(6 баллов)**

3. Автомат обрабатывает натуральное число N>1 по следующему алгоритму.

1. Строится двоичная запись числа N.

2. Последняя цифра записи удаляется.

3. Если исходное число N было нечетным, в конец записи (справа) дописываются цифры 10, если четным – 01.

4. Результат переводится в десятичную систему счисления и выводится на экран.

Какое число нужно ввести в автомат, чтобы в результате получилось 2022?

**(6 баллов)**

4. Для какого наименьшего целого неотрицательного числа A выражение (2x + 3y > 30) ∨ (x + y ≤ A) тождественно истинно, т.е. принимает значение 1 при любых целых неотрицательных x и y?

**(7 баллов)**

5. Сколько различных решений имеет уравнение (K ∧ L) ∨ (M ∧ N) = 1, где K, L, M, N – логические переменные?

В ответе не нужно перечислять все различные наборы значений K, L, M и N, при которых выполнено данное равенство. В качестве ответа вам нужно указать только количество таких наборов.

**(7 баллов)**

6. Определите, что будет напечатано в результате работы следующего фрагмента программы:

|  |  |  |  |
| --- | --- | --- | --- |
| **Python** | **Паскаль** | **Си++** | **Алгоритмический язык** |
| s = 5  k = 0  **while** k < 15:      k += 2      s += k  **print**(s) | **var** k, s: **integer**;  **begin**         s:=5;         k:=0;  **while** k < 15 **do** **begin**              k:=k+2;              s:=s+k;  **end**;  **write**(s);  **end**. | #include <iostream>  using namespace std;  **int** main() **{**  **int** s, k;      s = 5, k = 0;  **while** (k < 15) **{**          k = k + 2;          s = s + k;  **}**      cout << s << endl;  **return** 0;  **}** | **алг**  **нач**  **цел** k, s      s := 5      k := 0  **нц** **пока** k < 15          k := k + 2          s := s + k  **кц**  **вывод** s  **кон** |

**(8 баллов)**

7. Чему будет равно значение, вычисленное алгоритмом при выполнении вызова F(5)?

|  |  |  |  |
| --- | --- | --- | --- |
| **Алгоритмический язык** | **Python** | **Паскаль** | **Си** |
| **алг** **цел** F(**цел** n)  **нач**  **если** n > 2  **то**      з**нач** := F(n - 1)+F(n - 2)      знач := 1  все  кон | **def** F(n):  **if** n > 2:  **return** F(n-1)+ F(n-2)  **else**: **return** 1 | **function** F(n: **integer**): **integer**;  **begin**  **if** n > 2 **then**          F := F(n - 1) + F(n - 2)  **else**          F := 1;  **end**; | **int** F(**int** n)  **{**  **if** (n > 2)  **return** F(n-1) + F(n-2);  **else** **return** 1;  **}** |

**(8 баллов)**

8. В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 5; 1; 6; 7; 8; 8; 7; 7; 6; 9 соответственно, т. е. A[0] = 5; A[1] = 1 и т. д. Определите значение переменной c после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

|  |  |  |  |
| --- | --- | --- | --- |
| **Алгоритмический язык** | **Python** | **Паскаль** | **Си++** |
| c := 0  **нц** **для** i **от** 1 **до** 9  **если** A[i - 1] >= A[i] **то**      t := A[i]      A[i] := A[i - 1]      A[i - 1] := t  **иначе**      c := c + 1  **все**  **кц** | c = 0  **for** i **in** range(1, 10):  **if** A[i - 1] >= A[i]:          t = A[i]          A[i] = A[i - 1]          A[i - 1] = t  **else**:          c = c + 1 | c := 0;  **for** i := 1 **to** 9 **do**  **if** A[i - 1] >= A[i] **then**  **begin**              t := A[i];              A[i] := A[i - 1];              A[i - 1] := t  **end**  **else**          c := c + 1; | c = 0;  **for** (i = 1; i <= 9; i++)  **if** (A[i-1] >= A[i])  **{**          t = A[i];          A[i] = A[i - 1];          A[i - 1] = t;  **}**  **else**          c++; |

**(8 баллов)**

9. Напишите в ответе наименьшее значение входной переменной k, при котором программа выдаёт ответ 21.

|  |  |  |  |
| --- | --- | --- | --- |
| **Алгоритмический язык** | **Python** | **Паскаль** | **Си++** |
| **алг**  **нач**  **цел** i, k  **ввод** k    i := 1  **нц** **пока** f(i) <= k\*g(i)      i := i + 1  **кц**  **вывод** i  **кон**    **алг** **цел** f(**цел** n)  **нач**    з**нач** := n \* n \* n  **кон**    **алг** **цел** g(**цел** n)  **нач**    з**нач** := n \* n  **кон** | **def** f(n):  **return** n\*n\*n    **def** g(n):  **return** n\*n    k = int(input())  i = 1  **while** f(i) <= k\*g(i):    i += 1  **print** (i) | **var**    k, i : longint;    **function** f(n: longint): longint;  **begin**    f := n \* n \* n;  **end**;    begin    g := n\*n;  end;    begin    readln(k);    i := 1;    while f(i) <= k\*g(i) do      i := i+1;    writeln(i)  end. | #include <iostream>  using namespace std;  **long** f(**long** n) **{**  **return** n \* n \* n;  **}**    **long** g(**long** n) **{**  **return** n\*n;  **}**    **int** main()  **{**  **long** k, i;    cin >> k;    i = 1;  **while** (f(i) <= k\*g(i))      i++;    cout << i << endl;  **return** 0;  **}** |

**(8 баллов)**

10. Дано целое число N (> 2) Реализовать алгоритм определения простоты числа N: число является простым, если у него всего два делителя: 1 и N.

**(12 баллов)**

11. Дан одномерный массив размера N (N — четное число). Элементы массива – целые числа. Найти номера двух ближайших элементов из этого массива (то есть элементов с наименьшим модулем разности значений) и вывести эти номера в порядке возрастания. Вспомогательный массив не использовать.

**(12 баллов)**

12. Компьютер наземной станции слежения получает от объектов-самолётов, находящихся в зоне её работы, идентификационные сигналы, представляющие собой **последовательность из N целых положительных чисел**. Каждый объект направляет на наземную станцию уникальное число, т. е. все числа в получаемой станцией последовательности **различны**. Обработка сигнала представляет собой рассмотрение **всех пар** различных элементов последовательности, при этом элементы пары не обязаны быть переданы непосредственно друг за другом, **порядок элементов в паре не важен**. Считается, что возникла одна критическая ситуация, если **произведение элементов некоторой пары кратно 58**.

*Необходимо определить общее количество возникших критических ситуаций.*

*Описание входных и выходных данных*

В первой строке входных данных задаётся количество чисел **N** (**1 < N < 1000**). В каждой из последующих N строк записано одно целое положительное число, **не превышающее 10 000**.

В качестве результата программа должна напечатать одно число: общее количество возникших критических ситуаций.

*Пример входных данных:*

4

2

6

29

87

*Пример выходных данных для приведённого выше примера входных данных:*

4

Из четырёх заданных чисел можно составить 6 попарных произведений:

2\*6 = 12

2\*29 = 58

2\*87 = 174

6\*29 = 174

6\*87 = 522

29\*87 = 2523

Из них на 58 делятся 4 произведения (выделены синим).

Требуется написать эффективную по времени и по памяти программу для решения описанной задачи.

**(12 баллов)**