

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Заболотни Галина Ивановна

Должность: Директор филиала

Дата подписания: 19.01.2023 13:07:12

Уникальный идентификатор:

476db7d4ac1eb3def8136172be235477473d63457266ce26b7e9e40f733b8b08

САМАРСКИЙ
ПОЛИТЕХ

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Самарский государственный технический университет»
(ФГБОУ ВО «СамГТУ»)

УТВЕРЖДАЮ

Ректор ФГБОУ ВО «СамГТУ»,
Д.Е. Быков

Д.Е. Быков

2021 г.



**ПРОГРАММА
ВСТУПИТЕЛЬНЫХ ИСПЫТАНИЙ
для абитуриентов, поступающих на базе СПО**

по направлениям подготовки

- 01.03.02 Прикладная математика и информатика;
- 09.03.01 Информатика и вычислительная техника;
- 09.03.02 Информационные системы и технологии
- 09.03.03 Прикладная информатика;
- 09.03.04 Программная инженерия;
- 10.03.01 Информационная безопасность;
- 11.03.01 Радиотехника;
- 12.03.01 Приборостроение;
- 15.03.04 Автоматизация технологических процессов и производств
- 27.03.03 Системный анализ и управление
- 27.03.04 Управление в технических системах
(профиль: «Управление и информатика в технических системах»)

код и наименование направления подготовки

по дисциплине

ОСНОВЫ ПРОГРАММИРОВАНИЯ

Самара, 2021

ОСНОВНЫЕ ПОНЯТИЯ И ФАКТЫ ДИСЦИПЛИНЫ ОСНОВЫ ПРОГРАММИРОВАНИЯ

1. ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ

Темы курса основной школы, рекомендованные для повторения:

Основные понятия алгоритмизации. Основные понятия алгоритмизации. Понятие алгоритма. Свойства алгоритма. Схема решения задач на ЭВМ. Формы записи алгоритмов. Общие принципы построения алгоритмов. Основные алгоритмические конструкции: линейные, разветвляющиеся, циклические. Логические основы алгоритмизации. Основные базовые и структурированные типы данных, их характеристика. Анализ алгоритмов. Оптимальные линейные программы. Робот. Исполнитель Чертёжник. Исполнитель Редактор. Введение в язык Python. Простейшая программа. Переменные. Типы данных. Размещение переменных в памяти. Арифметические выражения и операции. Вычисления. Деление нацело и остаток. Вещественные значения. Стандартные функции. Случайные числа. Ветвления. Условный оператор. Сложные условия. Циклические алгоритмы. Цикл с условием. Поиск максимальной цифры числа. Алгоритм Евклида. Циклы с постусловием. Циклы по переменной. Вложенные циклы. Процедуры. Процедуры с параметрами. Локальные и глобальные переменные. Функции. Вызов функции. Возврат нескольких значений. Логические функции. Рекурсия. Ханойские башни. Использование стека. Анализ рекурсивных функций. Массивы. Ввод и вывод массива. Перебор элементов. Алгоритмы обработки массивов. Поиск в массиве. Максимальный элемент. Реверс массива. Сдвиг элементов массива. Отбор нужных элементов. Сортировка массивов. Метод пузырька (сортировка обмёнами). Метод выбора. Сортировка слиянием. «Быстрая сортировка». Двоичный поиск. Символьные строки. Операции со строками. Поиск в строках. Примеры обработки строк. Преобразование число-строка. Строки в 11 процедурах и функциях. Рекурсивный перебор. Матрицы. Обработка элементов матрицы. Обработка массивов. Обработка строк.

2. ЯЗЫКИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ

Темы курса основной школы, рекомендованные для повторения:

Поколения языков программирования. Языки программирования. Эволюция языков программирования. Классификация языков программирования. Понятие системы программирования. Исходный, объектный и загрузочный модули. Интегрированная среда программирования. Методы программирования: структурный, модульный, объектноориентированный. Достоинства и

недостатки методов программирования. Общие принципы разработки программного обеспечения. Жизненный цикл программного обеспечения. Типы приложений. Консольные приложения.

3. ПРОГРАММИРОВАНИЕ НА АЛГОРИТМИЧЕСКОМ ЯЗЫКЕ ПАСКАЛЬ

Темы курса основной школы, рекомендованные для повторения:

Turbo Pascal. Основные элементы языка. Структурная схема программы на алгоритмическом языке. Лексика языка. Переменные и константы. Типы данных. Выражения и операции. Turbo Pascal. Стандартные функции. Структура программы. Операторы языка. Синтаксис операторов. Составной оператор. Вложенные условные операторы. Циклические конструкции. Turbo Pascal. Массивы как структурированный тип данных. Объявление массива. Ввод и вывод одномерных и двумерных массивов. Обработка массивов. Стандартные функции для массива целых и вещественных чисел. Объявление строковых типов данных. Поиск, удаление, замена и добавление символов в строке. Операции со строками. Стандартные функции и процедуры работы со строками.

4. ПРОГРАММИРОВАНИЕ В КОНСОЛЬНОЙ СРЕДЕ C++

Темы курса основной школы, рекомендованные для повторения:

Интегрированная среда разработки CodeBlocks. Интерфейс консольной среды программирования: характеристика, объекты. Панель компонентов. Окно кода проекта. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.

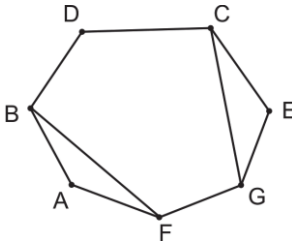
СТРУКТУРА БИЛЕТА И КРИТЕРИИ ОЦЕНКИ

Вступительное испытание оценивается по 100-бальной шкале. Минимальное количество баллов для получения оценки «зачтено» - 40.

Структура экзаменационного билета состоит из двух частей. Первая часть состоит из 16 заданий, за каждое правильно выполненное задание - 4 балла. Максимальное количество баллов за первую часть - 64.

Вторая часть состоит из 4 заданий, за каждое правильно выполненное задание - 9 баллов. Максимальное количество баллов за вторую часть – 36

ОБРАЗЕЦ ЭКЗАМЕНАЦИОННОГО ЛИСТА

Первая часть. Напишите ответ в области ответов I.																																																																				
№	Задание																																																																			
1	Миша заполнял таблицу истинности функции $(\neg x \wedge \neg y) \vee (y \equiv z) \vee \neg w$, но успел заполнить лишь фрагмент из трёх различных её строк, даже не указав, какому столбцу таблицы соответствует каждая из переменных w, x, y, z. <table><tr><td></td><td></td><td></td><td></td><td>$(\neg x \wedge \neg y) \vee (y \equiv z) \vee \neg w$</td></tr><tr><td>0</td><td></td><td>0</td><td>1</td><td>0</td></tr><tr><td></td><td>0</td><td></td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td></td><td>0</td></tr></table> Определите, какому столбцу таблицы соответствует каждая из переменных w, x, y, z. В ответе напишите буквы w, x, y, z в том порядке, в котором идут соответствующие им столбцы (сначала буква, соответствующая первому столбцу; затем буква, соответствующая второму столбцу, и т.д.). Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.								$(\neg x \wedge \neg y) \vee (y \equiv z) \vee \neg w$	0		0	1	0		0		1	0	0	1	1		0																																												
				$(\neg x \wedge \neg y) \vee (y \equiv z) \vee \neg w$																																																																
0		0	1	0																																																																
	0		1	0																																																																
0	1	1		0																																																																
2	На рисунке изображена схема дорог некоторого района. В соответствующей таблице символом “*” обозначено наличие дороги из одного населённого пункта в другой. Отсутствие этого символа обозначает, что такой дороги нет. Каждому населённому пункту на схеме соответствует его номер в таблице, но неизвестно, какой именно номер. Определите какие номера населённых пунктов в таблице могут соответствовать населённым пунктам Ви Сна схеме. В ответе запишите эти два номера в порядке их убывания без пробелов.  <table><tr><td></td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr><tr><td>1</td><td></td><td></td><td>*</td><td>*</td><td></td><td></td><td>*</td></tr><tr><td>2</td><td></td><td></td><td></td><td></td><td>*</td><td>*</td><td></td></tr><tr><td>3</td><td></td><td>*</td><td></td><td></td><td></td><td>*</td><td></td></tr><tr><td>4</td><td></td><td>*</td><td></td><td></td><td>*</td><td></td><td>*</td></tr><tr><td>5</td><td>*</td><td></td><td>*</td><td></td><td></td><td>*</td><td></td></tr><tr><td>6</td><td></td><td>*</td><td>*</td><td></td><td></td><td></td><td></td></tr><tr><td>7</td><td>*</td><td></td><td></td><td>*</td><td>*</td><td></td><td></td></tr></table>					1	2	3	4	5	6	7	1			*	*			*	2					*	*		3		*				*		4		*			*		*	5	*		*			*		6		*	*					7	*			*	*		
	1	2	3	4	5	6	7																																																													
1			*	*			*																																																													
2					*	*																																																														
3		*				*																																																														
4		*			*		*																																																													
5	*		*			*																																																														
6		*	*																																																																	
7	*			*	*																																																															
3	Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г, Д, Е решили использовать неравномерный двоичный код, удовлетворяющий условию Фано. Для буквы А использовали кодовое слово 0; для буквы Б – кодовое слово 10. Какова наименьшая возможная сумма длин кодовых слов для букв В, Г, Д, Е?																																																																			
4	На вход алгоритма подаётся натуральное число N. Алгоритм строит по нему число R следующим образом. 1) Строится двоичная запись числа N. 2) К этой записи дописываются справа ещё два разряда по следующему правилу: если N чётное, то справа от числа дописывается сначала 0, а затем 1. В противном случае, если N нечётное, то справа дописывается сначала 1, а затем 0. Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью числа R – результата работы																																																																			

	данного алгоритма. Укажите минимальное число R , которое больше 102 и может являться результатом работы данного алгоритма. В ответе число записать в десятичной системе счисления.								
5	Имеется набор данных, состоящий из пар положительных целых чисел. Необходимо выбрать из каждой пары ровно одно число так, чтобы сумма всех выбранных чисел не делилась на 3 и при этом была максимально возможной. Гарантируется, что искомую сумму получить можно. Программа должна напечатать одно число — максимально возможную сумму, соответствующую условиям задачи. Входные данные. Файл А Файл В Даны два входных файла (файл А и файл В), каждый из которых содержит в первой строке количество пар N ($1 \leq N \leq 100000$). Каждая из следующих N строк содержит два натуральных числа, не превышающих 10 000. Пример организации исходных данных во входном файле: <pre>6 1 3 5 12 6 9 5 4 3 3 1 1</pre> Для указанных входных данных значением искомой суммы должно быть число 32. В ответе укажите два числа: сначала значение искомой суммы для файла А, затем для файла В. Предупреждение: для обработки файла В не следует использовать переборный алгоритм, вычисляющий сумму для всех возможных вариантов, поскольку написанная по такому алгоритму программа будет выполняться слишком долго.								
6	Запишите число, которое будет напечатано в результате выполнения следующей программы. Программа представлена на пяти языках программирования. <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%; text-align: center;">Бейсик</th><th style="width: 50%; text-align: center;">Python</th></tr> </thead> <tbody> <tr> <td style="vertical-align: top;"> <pre>DIM S, N AS INTEGER S = 0 N = 75 WHILE S + N < 150 S = S + 15 N = N - 5 WEND PRINT N</pre> </td><td style="vertical-align: top;"> <pre>s = 0 n = 75 while s + n < 150: s = s + 15 n = n - 5 print(n)</pre> </td></tr> <tr> <th style="text-align: center;">Алгоритмический язык</th><th style="text-align: center;">Pascal</th></tr> <tr> <td style="vertical-align: top;"> <pre><u>алг</u> <u>нач</u> <u>целн</u>, s s := 0</pre> </td><td style="vertical-align: top;"> <pre>var s, n: integer; begin s := 0; n := 75;</pre> </td></tr> </tbody> </table>	Бейсик	Python	<pre>DIM S, N AS INTEGER S = 0 N = 75 WHILE S + N < 150 S = S + 15 N = N - 5 WEND PRINT N</pre>	<pre>s = 0 n = 75 while s + n < 150: s = s + 15 n = n - 5 print(n)</pre>	Алгоритмический язык	Pascal	<pre><u>алг</u> <u>нач</u> <u>целн</u>, s s := 0</pre>	<pre>var s, n: integer; begin s := 0; n := 75;</pre>
Бейсик	Python								
<pre>DIM S, N AS INTEGER S = 0 N = 75 WHILE S + N < 150 S = S + 15 N = N - 5 WEND PRINT N</pre>	<pre>s = 0 n = 75 while s + n < 150: s = s + 15 n = n - 5 print(n)</pre>								
Алгоритмический язык	Pascal								
<pre><u>алг</u> <u>нач</u> <u>целн</u>, s s := 0</pre>	<pre>var s, n: integer; begin s := 0; n := 75;</pre>								

	<pre> n := 75 нцпока s + n < 150 s := s + 15 n := n - 5 кц ВЫВОДn кон </pre>	<pre> while s + n < 150 do begin s := s + 15; n := n - 5; end; writeln(n) end. </pre>
	C++	
	<pre> #include <iostream> using namespace std; int main() { int s = 0, n = 75; while (s + n < 150) { s = s + 15; n = n - 5; } cout << n << endl; return 0; } </pre>	
7	Сергей составляет только 5-буквенные слова, в которых есть только буквы О, В, Е, Н, причём в каждом слове есть ровно одна гласная буква и она встречается ровно 1 раз. Каждая из допустимых согласных букв может встречаться в слове любое количество раз или не встречаться вовсе. Словом считается любая допустимая последовательность букв, не обязательно осмысленная. Сколько существует таких слов, которые может записать Сергей?	
8	Автоматическая камера производит растровые изображения размером 200x256 пикселей. Для кодирования цвета каждого пикселя используется одинаковое количество бит, коды пикселей записываются в файл один за другим без промежутков. Объём файла с изображением не может превышать 70 Кбайт без учёта размера заголовка файла. Какое максимальное количество цветов можно использовать в палитре?	
9	Набор данных состоит из нечётного количества пар натуральных чисел. Необходимо выбрать из каждой пары ровно одно число так, чтобы чётность суммы выбранных чисел совпадала с чётностью большинства выбранных чисел и при этом сумма выбранных чисел была как можно меньше. Определите минимальную сумму, которую можно получить при таком выборе. Гарантируется, что удовлетворяющий условиям выбор возможен. Входные данные. Файл А Файл В Первая строка входного файла содержит число N — общее количество пар в наборе. Каждая из следующих N строк содержит два натуральных числа, не превышающих 10 000. Пример входного файла: <pre> 5 15 8 5 11 6 3 7 2 9 14 </pre>	

	Для указанных данных надо выбрать числа 8, 5, 3, 2 и 9. Большинство из них нечётны, сумма выбранных чисел равна 27 и тоже нечётна. В ответе надо записать число 27. Вам даны два входных файла (А и В), каждый из которых имеет описанную выше структуру. В ответе укажите два числа: сначала значение искомой суммы для файла А, затем для файла В. Предупреждение: для обработки файла В не следует использовать переборный алгоритм, вычисляющий сумму для всех возможных вариантов, поскольку написанная по такому алгоритму программа будет выполняться слишком долго.
10	При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 6 символов и содержащий только символы из 32-символьного набора прописных латинских букв. В базе данных для хранения сведений о каждом пользователе отведено одинаковое и минимально возможное целое число байт. При этом используют посимвольное кодирование паролей, все символы кодируют одинаковым и минимально возможным количеством бит. Кроме пароля для каждого пользователя в системе хранятся дополнительные сведения, для чего выделено целое число байт; это число одно и то же для всех пользователей. Для хранения сведений о 30 пользователях потребовалось 600 байт. Сколько байт выделено для хранения дополнительных сведений об одном пользователе? В ответе запишите только целое число – количество байт.
11	Некоторый алгоритм получает на вход строку цифр и преобразовывает её. В алгоритме существует только две команды, в каждой из которых х и у обозначают последовательность цифр. 1) заменить (х,у). Эта команда заменяет в исходной строке первое слева вхождение цепочки х на цепочку у. Если в строке нет вхождений цепочки х, то выполнение команды заменить(х,у) не меняет эту строку. 2) нашлось(х) Эта команда проверяет, встречается ли цепочка х в исходной строке. Если она встречается, то команда возвращает логическое значение «истина», в противном случае она возвращает значение «ложь». Исходная строка при этом не меняется. Какая строка получится в результате примененного ниже алгоритма к строке, состоящей из 84 подряд идущих цифр 1? В ответе запишите полученную строку. НАЧАЛО ПОКА нашлось(11111) ИЛИ нашлось(888) ЕСЛИ нашлось(11111) ТО заменить(11111,88) ИНАЧЕ ЕСЛИ нашлось(888) ТО заменить(888,8) КОНЕЦ ЕСЛИ КОНЕЦ ЕСЛИ КОНЕЦ ПОКА КОНЕЦ
12	Значение арифметического выражения $9^7 + 3^{21} - 9$ записали в системе счисления с основанием 3. Сколько цифр «2» содержится в этой записи?
13	Исполнитель Робот ходит по клеткам бесконечной вертикальной клетчатой доски, переходя по одной из команд вверх, вниз, вправо, влево в

	соседнюю клетку в указанном направлении. Робот выполнил следующую программу: вверх влево влево вниз вниз вправо вправо вниз вправо вверх Укажите наименьшее возможное число команд в программе, переводящей Робота из той же начальной клетки в ту же конечную.														
14	В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ « », а для обозначения логической операции «И» - символ «&». В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет. <table border="1"> <thead> <tr> <th>Запрос</th><th>Найдено страниц (в сотнях тысяч)</th></tr> </thead> <tbody> <tr> <td>Горло</td><td>45</td></tr> <tr> <td>Корабль</td><td>40</td></tr> <tr> <td>Нос</td><td>30</td></tr> <tr> <td>Корабль & Нос</td><td>25</td></tr> <tr> <td>Горло & Нос</td><td>10</td></tr> <tr> <td>Горло & Корабль</td><td>0</td></tr> </tbody> </table> Какое количество страниц (в сотнях тысяч) будет найдено по запросу Горло Корабль Нос?	Запрос	Найдено страниц (в сотнях тысяч)	Горло	45	Корабль	40	Нос	30	Корабль & Нос	25	Горло & Нос	10	Горло & Корабль	0
Запрос	Найдено страниц (в сотнях тысяч)														
Горло	45														
Корабль	40														
Нос	30														
Корабль & Нос	25														
Горло & Нос	10														
Горло & Корабль	0														
15	Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_7, y_1, y_2, \dots, y_7$, которые удовлетворяют всем перечисленным ниже условиям? $(y_1 \rightarrow (y_2 \wedge x_1)) \wedge (x_1 \rightarrow x_2) = 1$ $(y_2 \rightarrow (y_3 \wedge x_2)) \wedge (x_2 \rightarrow x_3) = 1$... $(y_6 \rightarrow (y_7 \wedge x_6)) \wedge (x_6 \rightarrow x_7) = 1$ $y_7 \rightarrow x_7 = 1$ В ответе не нужно перечислять различные наборы значений переменных, при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких наборов.														
16	На вход алгоритма подаётся натуральное число N. Алгоритм строит по нему новое число следующим образом. 1) Строится двоичная запись числа N. 2) К этой записи дописываются справа ещё два разряда по следующему правилу: если N чётное, в конец числа (справа) дописываются два нуля, в противном случае справа дописываются две единицы. Например, двоичная запись 1001 числа 9 будет преобразована в 100111. Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью числа — результата работы данного алгоритма.														

	Укажите максимальное число N , для которого результат работы алгоритма будет меньше 134. В ответе это число запишите в десятичной системе счисления.	
Вторая часть. Напишите ответ в области ответов 2.		
1	На обработку поступает натуральное число, не превышающее 10^9 . Нужно написать программу, которая выводит на экран минимальную чётную цифру этого числа. Если в числе нет чётных цифр, то требуется вывести на экран «NO». Программист написал программу неправильно. Ниже эта программа представлена на пяти языках программирования.	
	Бейсик	Python
	<pre>DIM N, DIGIT, MINDIGIT AS LONG INPUT N MINDIGIT = N MOD 10 WHILE N > 0 DIGIT = N MOD 10 IF DIGIT MOD 2 = 0 THEN IF DIGIT < MINDIGIT THEN MINDIGIT = DIGIT END IF END IF N=N \ 10 WEND IF MINDIGIT = 0 THEN PRINT "NO" ELSE PRINT MINDIGIT END IF</pre>	<pre>N = int (input()) minDigit = N % 10 while N > 0: digit = N % 10 if digit % 2 == 0: if digit < minDigit: minDigit = digit N = N // 10 if minDigit == 0: print ("NO") else: print (minDigit)</pre>
	Алгоритмический язык	Pascal
	<pre>алг нач целN, digit, minDigit вводN minDigit := mod(N,10) нцпокаN > 0 digit := mod(N,10) еслиmod(digit, 2) = 0 то еслиdigit < minDigit то minDigit := digit все все N := div(N,10) кц еслиminDigit = 0 то вывод"NO" иначе выводminDigit все кон</pre>	<pre>var N, digit, minDigit: longint; begin readln(N); minDigit := N mod 10; while N > 0 do begin digit := N mod 10; if digit mod 2 = 0 then if digit < minDigit then minDigit := digit; N := N div 10; end; if minDigit = 0 then writeln ("NO") else writeln(minDigit) end. end.</pre>
	C++	
	<pre>#include <iostream> using namespace std;</pre>	

	<pre> int main() { int N, digit, minDigit; cin >> N; minDigit = N % 10; while (N > 0) { digit = N % 10; if (digit % 2 == 0) if (digit < minDigit) minDigit = digit; N = N / 10; } if (minDigit == 0) cout << "NO" << endl; else cout << minDigit << endl; return 0; } </pre>
	Последовательно выполните следующее. 1. Напишите, что выведет программа при вводе числа 231. 2. Найдите допущенные программистом ошибки и исправьте их. Исправление ошибки должно затрагивать только строку, в которой находится ошибка. Для каждой ошибки необходимо выписать строку, в которой сделана ошибка и указать, как исправить ошибку, т.е. привести правильный вариант строки. Известно, что в тексте программы можно исправить ровно две строки так, чтобы она работала правильно. Достаточно указать ошибки и способ их исправления для одного языка программирования. Обратите внимание на то, что требуется найти ошибки в имеющейся программе, а не написать свою, возможно, использующую другой алгоритм решения.
2	Дан целочисленный массив из 30 элементов. Элементы массива могут принимать натуральные значения от 1 до 10000 включительно. Опишите на одном из языков программирования алгоритм, который находит минимум среди элементов массива, не делящихся нацело на 6, а затем заменяет каждый элемент, не делящийся нацело на 6, на число, равное найденному минимуму. Гарантируется, что хотя бы один такой элемент в массиве есть. В качестве результата необходимо вывести изменённый массив, каждый элемент выводится с новой строки. Например, для исходного массива из шести элементов: <pre> 14 6 11 18 9 24 </pre> программа должна вывести следующий массив: <pre> 9 6 9 18 9 </pre>

	24 В качестве ответа нужно привести программу, написанную на любом из языков программирования, но необходимо указать название данного языка.
3	На вход программы поступает последовательность из N целых положительных чисел, все числа в последовательности различны. Рассматриваются все пары различных элементов последовательности, находящихся на расстоянии не менее чем 4 (разница в индексах элементов пары должна быть 4 или более, порядок элементов в паре неважен). Необходимо определить количество таких пар, для которых произведение элементов делится на 27. В первой строке входных данных задаётся количество чисел N ($4 \leq N \leq 1000$). В каждой из последующих N строк записано одно целое положительное число, не превышающее 10000. В качестве результата программа должна вывести одно число: количество пар элементов, находящихся в последовательности на расстоянии не меньше, чем 4, в которых произведение элементов кратно 29. В качестве ответа нужно привести программу, написанную на любом из языков программирования, но необходимо указать название данного языка.
4	Исполнитель «Вычислитель» преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера: 1. Прибавить 2 2. Умножить на 2 3. Прибавить 3 Программа для «Вычислителя» — это последовательность команд. Сколько существует таких программ, которые преобразуют исходное число 2 в число 22 и при этом траектория вычислений программы содержит число 11? Траектория – это последовательность результатов выполнения всех команд программы.

СПИСОК ЛИТЕРАТУРЫ

1. Н.Д. Угринович Информатика и ИКТ: учебник 10-11 класс - М.: БИНОМ, 2016 - 212 с.
2. Семакин И.Г., Хеннер Е.К., Шеина Т.Ю. Информатика. Базовый уровень: учебник для 11 класса. - 3-е изд. - М.: БИНОМ. Лаборатория знаний, 2014. -224 с.
3. Поляков К.Ю., Еремин Е.А. Информатика, 11 класс. Углубленный уровень: учебник для 11 класса: в 2 ч. 4.1. - М.: БИНОМ. Лаборатория знаний, 2013. - 240 с.
4. Культин Н.Б. Программирование вTurboPascal 7.0 и Delphi. СПб.: БХВ-Петербург, 2012.
5. Вовк Е.Т. Информатика. Пособие для подготовки к ЕГЭ. Учебно-методическое пособие. - М.: БИНОМ. Лаборатория знаний, 2018.
6. Страуструп Б. Язык программирования Turbo Pascal (третье издание). – Спб., М.: "Невский диалект. Издательство "Бином", 2010.
7. Эпштейн М.С. Практикум по программированию: учебное пособие для сред. проф. образования. – М.: Издательский центр «Академия», 2009.
8. Поляков К.Ю., Еремин Е.А. Информатика, 10 класс. Углубленный уровень- М.: БИНОМ, 2015.