

РАБОЧАЯ ПРОГРАММА
по курсу внеурочной деятельности
общеинтеллектуального направления
«Основы программирования»
для 11 класса

Составитель:
Самсонова Ирина Леонидовна
первая квалификационная категория

Пояснительная записка

Рабочая программа по курсу внеурочной деятельности «Основы программирования» общеинтеллектуального направления для 11 класса разработана в соответствии с:

- Федеральным законом от 29 декабря 2012 года №273-ФЗ «Об образовании в Российской Федерации»;
- Законом Архангельской области от 02 июля 2013 года №712-41-ОЗ «Об образовании в Архангельской области»;
- Требованиями Федерального Государственного образовательного стандарта основного общего образования, утвержденного приказом Министерства образования РФ от 17 декабря 2010 г. №1887;
- Письмом Министерства образования и науки РФ «Об организации внеурочной деятельности при введении федерального государственного образовательного стандарта общего образования» № 03-296 от 12 мая 2011 года;
- Требованиями к результатам освоения основной образовательной программы основного общего образования.

Курс внеурочной деятельности по предмету информатика «Основы программирования» рассчитан на один учебный год (68 часов; 2 часа в неделю).

РАЗДЕЛ I. РЕЗУЛЬТАТЫ ОСВОЕНИЯ КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ «ОСНОВЫ ПРОГРАММИРОВАНИЯ»

Личностные результаты:

- 1) сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- 2) готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности;
- 3) навыки сотрудничества со сверстниками, детьми младшего возраста, взрослыми в образовательной, учебно-исследовательской, проектной и других видах деятельности;
- 4) эстетическое отношение к миру, включая эстетику научного и технического творчества;
- 5) осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов; отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

Метапредметные результаты:

- 1) умение самостоятельно определять цели деятельности и составлять планы деятельности; самостоятельно осуществлять, контролировать и корректировать деятельность; использовать все возможные ресурсы для достижения поставленных целей и реализации планов деятельности; выбирать успешные стратегии в различных ситуациях;
- 2) умение продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции других участников деятельности, эффективно разрешать конфликты;
- 3) владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем; способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания;
- 4) готовность и способность к самостоятельной информационно-познавательной деятельности, включая умение ориентироваться в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников;
- 5) умение использовать средства информационных и коммуникационных технологий в решении когнитивных, коммуникативных и организационных задач с соблюдением

требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности.

Предметные результаты:

- 1) сформированность представлений о роли информации и связанных с ней процессов в окружающем мире;
- 2) владение системой базовых знаний, отражающих вклад информатики в формирование современной научной картины мира;
- 3) сформированность представлений о важнейших видах дискретных объектов и об их простейших свойствах, алгоритмах анализа этих объектов, о кодировании и декодировании данных и причинах искажения данных при передаче;
- 4) систематизация знаний, относящихся к математическим объектам информатики; умение строить математические объекты информатики, в том числе логические формулы;
- 5) сформированность базовых навыков и умений по соблюдению требований техники безопасности, гигиены и ресурсосбережения при работе со средствами информатизации;
- 6) сформированность представлений об устройстве современных компьютеров, о тенденциях развития компьютерных технологий; о понятии «операционная система» и основных функциях операционных систем; об общих принципах разработки и функционирования интернет-приложений;
- 7) сформированность представлений о компьютерных сетях и их роли в современном мире; знаний базовых принципов организации и функционирования компьютерных сетей, норм информационной этики и права, принципов обеспечения информационной безопасности, способов и средств обеспечения надёжного функционирования средств ИКТ;
- 8) понимания основ правовых аспектов использования компьютерных программ и работы в Интернете;
- 9) владение опытом построения и использования компьютерно-математических моделей, проведения экспериментов и статистической обработки данных с помощью компьютера, интерпретации результатов, получаемых в ходе моделирования реальных процессов; умение оценивать числовые параметры моделируемых объектов и процессов; сформированность представлений о необходимости анализа соответствия модели и моделируемого объекта (процесса);
- 10) сформированность представлений о способах хранения и простейшей обработке данных; умение пользоваться базами данных и справочными системами; владение основными сведениями о базах данных, их структуре, средствах создания и работы с ними;
- 11) владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов;
- 12) овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки;
- 13) владение стандартными приёмами написания на алгоритмическом языке программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ; использование готовых прикладных компьютерных программ по выбранной специализации;
- 14) владение универсальным языком программирования высокого уровня (по выбору), представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции;
- 15) владение умением понимать программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня; знанием основных конструкций программирования; умением анализировать алгоритмы с использованием таблиц;
- 16) владение навыками и опытом разработки программ в выбранной среде программирования, включая тестирование и отладку программ; владение элементарными навыками формализации прикладной задачи и документирования программ.

РАЗДЕЛ II. СОДЕРЖАНИЕ КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ С УКАЗАНИЕМ ФОРМ ОРГАНИЗАЦИИ И ВИДОВ ДЕЯТЕЛЬНОСТИ

Каждое занятие может быть построено по следующему алгоритму:

1. повторение основных методов решения заданий по теме,
2. совместное решение заданий ЕГЭ,
3. самостоятельная работа обучающихся по решению заданий с хронометражем.

Раздел 1. «Контрольно-измерительные материалы ЕГЭ по информатике» - 2 часа

ЕГЭ как форма независимой оценки уровня учебных достижений выпускников. Особенности проведения ЕГЭ по информатике. Специфика формы контроля. Структура и содержание КИМов по информатике. Основные термины ЕГЭ.

Раздел 2. «Пользовательский курс» - 8 часов

Основные понятия классификации программного обеспечения, свойств и функциональных возможностей основных видов программного обеспечения, структуры файловой системы, включая правила именования каталогов и файлов. Основные правила адресации ячеек в электронной таблице. Понятие абсолютной и относительной адресации. Повторение принципов организации табличных (реляционных) баз данных. Технология адресации и поиска информации в Интернете.

Раздел 3. «Информация и системы счисления» - 10 часов

Повторение методов решения задач по теме. Решение тренировочных задач на измерение количества информации (вероятностный подход), кодирование текстовой информации и измерение ее информационного объема, кодирование графической информации и измерение ее информационного объема, кодирование звуковой информации и измерение ее информационного объема, умение кодировать и декодировать информацию. Понятие системы счисления, перевод чисел из одной системы счисления в другую. Умение осуществлять арифметические действия в различных системах счисления.

Раздел 4. «Алгоритмизация» - 10 часов

Формальное исполнение алгоритма, записанного на естественном языке. Создание линейного алгоритма для формального исполнителя с ограниченным набором команд. Исполнение алгоритма для конкретного исполнителя с фиксированным набором команд. Умение анализировать результат исполнения алгоритма.

Раздел 5. «Основы программирования» - 20 часов

Основные конструкции языка программирования, понятия переменной, оператора присваивания. Рекурсивный алгоритм. Работа с массивами (заполнение, считывание, поиск, сортировка, массовые операции и др.). Анализ алгоритма, содержащего вспомогательные алгоритмы, цикл и ветвление. Анализ программы, использующей процедуры и функции.

Чтение фрагмента программы на языке программирования и исправление допущенных ошибок. Построение дерева игры по заданному алгоритму и обоснование выигрышной стратегии

Решение тренировочных задач на поиск и исправление ошибок в небольшом фрагменте программы. Решение задач средней сложности на составление собственной эффективной программы (30-50 строк).

Раздел 6. «Логика» - 10 часов

Основные понятия и определения (таблицы истинности) трех основных логических операций (инверсия, конъюнкция, дизъюнкция), а также импликации. Повторение методов решения задач по теме. Решение тренировочных задач на построение и преобразование логических выражений, построение таблиц истинности, построение логических схем. Решение логических задач на применение основных законов логики при работе с логическими выражениями.

Раздел 7. «Тренинг по вариантам» - 8 часов

РАЗДЕЛ III. ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ урока	Тема	Кол-во часов	Дата	Факт
Контрольно-измерительные материалы по ЕГЭ – 2 часа				
1.	Особенности проведения ЕГЭ по информатике текущего года.	1		
2.	Структура и содержание КИМов по информатике.	1		
Пользовательский курс – 8 часов				
3.	Анализ информационных моделей.	1		
4.	Анализ информационных моделей.	1		
5.	Базы данных. Файловая система	1		
6.	Базы данных. Файловая система.	1		
7.	Электронные таблицы.	1		
8.	Электронные таблицы.	1		
9.	Поиск путей в графе.	1		
10.	Поиск информации. Работа с текстовым редактором.	1		
Информация и системы счисления – 10 часов				
11.	Системы счисления.	1		
12.	Кодирование чисел.	1		
13.	Кодирование и декодирование информации. Расшифровка сообщений. Выбор кода.	1		
14.	Кодирование и декодирование информации. Расшифровка сообщений. Выбор кода.	1		
15.	Перебор слов и системы счисления.	1		
16.	Перебор слов и системы счисления.	1		
17.	Кодирование и декодирование информации. Передача информации.	1		
18.	Кодирование и декодирование информации. Передача информации.	1		
19.	Вычисление количества информации.	1		
20.	Вычисление количества информации.	1		
Алгоритмизация – 10 часов				
21.	Анализ и построение алгоритмов для исполнителей.	1		
22.	Нестандартные исполнители.	1		
23.	Выполнение алгоритма для исполнителя Редактор.	1		
24.	Выполнение алгоритма для исполнителя Редактор.	1		
25.	Траектория вычислений программы.	1		
26.	Траектория вычислений программы.	1		
27.	Тренинг	1		
28.	Тренинг	1		
29.	Тренинг	1		
30.	Тренинг	1		
Основы программирования – 20 часов				
31.	Анализ программ.	1		
32.	Анализ программ.	1		
33.	Рекурсивные алгоритмы	1		
34.	Вызов рекурсивных процедур.	1		
35.	Наибольшее и наименьшее значение функции.	1		
36.	Алгоритмы с использованием условного оператора. Выбор чисел по заданному условию.	1		
37.	Динамическое программирование.	1		
38.	Динамическое программирование.	1		

39.	Динамическое программирование.	1		
40.	Динамическое программирование.	1		
41.	Алгоритмы, печатающие числа, наибольшее значение вводимого числа	1		
42.	Алгоритмы, печатающие числа, наименьшее значение вводимого числа	1		
43.	Перебор вариантов.	1		
44.	Алгоритмы обработки массивов	1		
45.	Алгоритмы обработки массивов	1		
46.	Обработка символьных строк	1		
47.	Обработка символьных строк	1		
48.	Обработка символьных строк	1		
49.	Тренинг	1		
50.	Тренинг	1		
Логика – 10 часов				
51.	Построение таблиц истинности логических выражений	1		
52.	Построение таблиц истинности логических выражений	1		
53.	Построение таблиц истинности логических выражений	1		
54.	Преобразование логических выражений. Логические высказывания	1		
55.	Преобразование логических выражений. Числовые отрезки	1		
56.	Стратегия игр.	1		
57.	Стратегия игр.	1		
58.	Выигрышная стратегия.	1		
59.	Выигрышная стратегия.	1		
60.	Выигрышная стратегия.	1		
Тренинг – 8 часов				
61.	Тренинг по вариантам с последующим разбором результатов	1		
62.	Тренинг по вариантам с последующим разбором результатов	1		
63.	Тренинг по вариантам с последующим разбором результатов	1		
64.	Разбор результатов	1		
65.	Тренинг по вариантам с последующим разбором результатов	1		
66.	Тренинг по вариантам с последующим разбором результатов	1		
67.	Тренинг по вариантам с последующим разбором результатов	1		
68.	Разбор результатов	1		

Промежуточная аттестация

1. Сколько существует целых чисел x , для которых выполняется неравенство $AB_{16} \leq x < 311_8$?

В ответе укажите количество чисел, сами числа писать не нужно.

2. Логическая функция F задаётся выражением $(x \equiv (w \vee y)) \vee ((w \rightarrow z) \wedge (y \rightarrow w))$.

Дан частично заполненный фрагмент, содержащий **неповторяющиеся** строки таблицы истинности функции F .

Определите, какому столбцу таблицы истинности соответствует каждая из переменных x, y, z, w .

Переменная 1	Переменная 2	Переменная 3	Переменная 4	Функция
???	???	???	???	F
1			1	0
			1	0
1		1		0

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.

Пример. Пусть задано выражение $x \rightarrow y$, зависящее от двух переменных x и y , и фрагмент таблицы истинности:

Переменная 1	Переменная 1	Функция
???	???	F
0	1	0

Тогда первому столбцу соответствует переменная y , а второму столбцу соответствует переменная x . В ответе нужно написать: yx .

3. Между населёнными пунктами А, В, С, D, E, F, G построены дороги, протяжённость которых приведена в таблице. Отсутствие числа в таблице означает, что прямой дороги между пунктами нет.

	A	B	C	D	E	F	G
A		2		6			
B	2		5	2			
C		5		4			8
D	6	2	4		2	7	
E				2			5
F				7			7
G			8		5	7	

Определите длину кратчайшего пути между пунктами А и G. Передвигаться можно только по указанным дорогам.

4. Даны фрагменты двух таблиц из базы данных. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информация представлена значением поля ID в соответствующей строке таблицы 1. На основании имеющихся данных определите ID человека, у которого в самом молодом возрасте появился первый правнук или правнучка. При вычислении ответа учитывайте только информацию из приведённых фрагментов таблиц.

Таблица 1				Таблица 2	
ID	Фамилия И.О.	Пол	Год рождения	ID Родителя	ID Ребенка
152	Павленко А. К.	М	1942	152	314
232	Сокол Е. П.	Ж	1964	232	803
314	Хитрук Е. А.	Ж	1970	314	468
323	Кривич Л. П.	Ж	1947	323	314
343	Симонян А. А.	М	1989	343	957
407	Хитрук П. А.	М	1937	407	760
424	Косых В. Г.	М	1984	407	232
468	Симонян С. И.	Ж	1992	424	880
613	Хитрук Н. П.	Ж	1939	468	957
760	Хитрук И. П.	М	1968	613	760
803	Сокол Л. М.	Ж	1986	613	232
880	Косых Г. В.	М	2007	760	468
902	Сокол М. Л.	М	1965	803	880
957	Симонян Т. А.	М	2017	902	803

5. По каналу связи передаются сообщения, содержащие только восемь букв: А, В, Е, З, И, Н, О, Р. Для передачи используется двоичный код, удовлетворяющий условию Фано. Кодовые слова для некоторых букв известны: А — 101, В — 010, И — 00. Какое наименьшее количество двоичных знаков потребуется для кодирования слова НЕВЕЗЕНИЕ?

Примечание. Условие Фано означает, что ни одно кодовое слово не является началом другого кодового слова.

6. На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом.

1. Строится двоичная запись числа N .
2. Далее эта запись обрабатывается по следующему правилу:
 - а) если число чётное, то к двоичной записи числа слева дописывается 1, а справа 0. Например, если для исходного числа 100 результатом будет являться число 11000;
 - б) если число нечётное, то к двоичной записи числа слева дописывается 11 и справа дописывается 11.

Полученная таким образом запись является двоичной записью искомого числа R .

Укажите минимальное число N , после обработки которого с помощью этого алгоритма получается число, большее, чем 52. В ответе запишите это число в десятичной системе счисления.

7. В ячейки диапазона C3:F6 электронной таблицы записаны числа, как показано на рисунке.

	A	B	C	D	E	F
1						
2						
3			1	2	3	4
4			11	13	15	17
5			21	24	27	30
6			31	35	39	43

В ячейке A1 записали формулу =E\$5-\$D4. После этого ячейку A1 скопировали в ячейку B2. Какое число будет показано в ячейке B2? Примечание: знак \$ используется для обозначения абсолютной адресации.

8. Определите, что будет напечатано в результате выполнения программы, записанной ниже на разных языках программирования.

Бейсик	Python
<pre> DIM N, S AS INTEGER N = 1 S = 0 WHILE N <= 650 S = S + 20 N = N * 5 WEND PRINT S </pre>	<pre> n = 1 s = 0 while n <= 650: s = s + 20 n = n * 5 print(s) </pre>
Паскаль	Алгоритмический язык
<pre> var n, s: integer; begin n := 1; s := 0; while n <= 650 do begin s := s + 20; n := n * 5 end; write(s) end. </pre>	<pre> алг нач цел n, s n := 1 s := 0 нц пока n <= 650 s := s + 20 n := n * 5 кц вывод s кон </pre>
Си++	
<pre> #include <iostream> </pre>	

```

using namespace std;
int main()
{
    int n, s;
    n = 1;
    s = 0;
    while (n <= 650)
    {
        s = s + 20;
        n = n * 5;
    }
    cout << s << endl;
    return 0;
}

```

9. Для хранения в информационной системе документы сканируются с разрешением 600 dpi и цветовой системой, содержащей $2^{24} = 16\,777\,216$ цветов. Методы сжатия изображений не используются. Средний размер отсканированного документа составляет 12 Мбайт. В целях экономии было решено перейти на разрешение 300 dpi и цветовую систему, содержащую $2^{16} = 65\,536$ цветов. Сколько Мбайт будет составлять средний размер документа, отсканированного с изменёнными параметрами?

10. Иван составляет 5-буквенные коды из букв И, В, А, Н. Буквы в коде могут повторяться, использовать все буквы не обязательно, но букву И нужно использовать хотя бы один раз. Сколько различных кодов может составить Иван?

11. Ниже на пяти языках программирования записан рекурсивный алгоритм F.

Бейсик	Python
<pre> SUB F(n) IF n > 0 THEN F(n \ 3) PRINT N F(n - 3) END IF END SUB </pre>	<pre> def F(n): if n > 0: F(n // 3) print(n) F(n - 3) </pre>
Паскаль	Алгоритмический язык
<pre> procedure F(n: integer); begin if n > 0 then begin F(n div 3); write(n); F(n - 3); end end; </pre>	<pre> алг F(цел n) нач если n > 0 то F(div(n,3)) вывод n F(n - 3) все кон </pre>
C++	
<pre> void F (int n) { if (n > 0) { F (n / 3); </pre>	

```
std::cout << n;
F (n - 3);
}
}
```

Запишите подряд без пробелов и разделителей все числа, которые будут напечатаны на экране при выполнении вызова F(9). Числа должны быть записаны в том же порядке, в котором они выводятся на экран.

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. При этом в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места — нули. Обычно маска записывается по тем же правилам, что и IP-адрес — в виде четырёх байтов, причём каждый байт записывается в виде десятичного числа. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.

Например, если IP-адрес узла равен 231.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 231.32.240.0.

Узлы с IP-адресами 98.162.71.151 и 98.162.71.155 находятся в разных сетях. Чему равно наименьшее количество возможных единиц в масках этих сетей?

13. Каждый сотрудник предприятия получает электронный пропуск, на котором записаны личный код сотрудника, номер подразделения и некоторая дополнительная информация. Личный код состоит из 11 символов, каждый из которых может быть русской буквой (используется 28 различных букв, каждая буква может быть заглавной или строчной) или одной из цифр от 1 до 9 (ноль для записи кодов не используется). Для записи кода на пропуске отведено минимально возможное целое число байт. При этом используют посимвольное кодирование, все символы кодируют одинаковым минимально возможным количеством бит. Номер подразделения — целое число от 1 до 700, он записан на пропуске как двоичное число и занимает минимально возможное целое число байт. Всего на пропуске хранится 30 байт данных. Сколько байт выделено для хранения дополнительных сведений об одном сотруднике? В ответе запишите только целое число — количество байт.

14. Исполнитель Редактор получает на вход строку цифр и преобразует её. Редактор может выполнять две команды, в обеих командах *v* и *w* обозначают цепочки цифр.

А) заменить (*v*, *w*).

Эта команда заменяет в строке первое слева вхождение цепочки *v* на цепочку *w*. Например, выполнение команды заменить (111, 27) преобразует строку 05111150 в строку 0527150.

Если в строке нет вхождений цепочки *v*, то выполнение команды заменить (*v*, *w*) не меняет эту строку.

Б) нашлось (*v*).

Эта команда проверяет, встречается ли цепочка *v* в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь». Строка исполнителя при этом не изменяется.

Цикл
ПОКА условие
 последовательность команд
КОНЕЦ ПОКА
выполняется, пока условие истинно.

Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 84 единиц?

НАЧАЛО

ПОКА нашлось (11111)

 заменить (222, 1)

 заменить (111, 2)

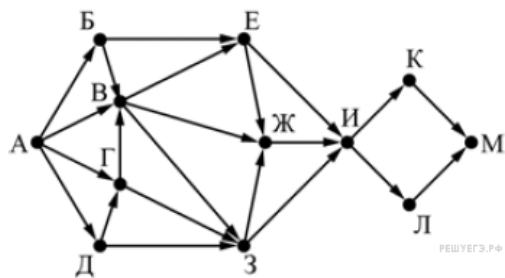
КОНЕЦ ПОКА

КОНЕЦ

15. На рисунке представлена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З, И, К, Л, М.

По каждой дороге можно двигаться только в одном направлении, указанном стрелкой.

Сколько существует различных путей из города А в город М, проходящих через город В?



16. Укажите наименьшее основание системы счисления, в которой запись числа 50 трехзначна.

17. 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.

Запрос	Найдено страниц (в сотнях тысяч)
Ухо	35
Подкова	25
Наковальня	40
Ухо Подкова Наковальня	70
Ухо & Наковальня	10
Ухо & Подкова	0

Какое количество страниц (в сотнях тысяч) будет найдено по запросу Подкова & Наковальня?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

18. На числовой прямой задан отрезок А. Известно, что формула $((x \in A) \rightarrow (x^2 \leq 81)) \wedge ((y^2 \leq 36) \rightarrow (y \in A))$

тождественно истинна при любых вещественных x и y . Какую наименьшую длину может иметь отрезок А?

19. Представленный ниже на пяти языках программирования фрагмент программы обрабатывает элементы одномерного целочисленного массива А с индексами от 0 до 9. Перед началом выполнения данного фрагмента эти элементы массива имели значения 2, 3,

4, 4, 10, 4, 5, 6, 12, 9 (т. е. $A[0] = 2, A[1] = 3, \dots, A[9] = 9$). Определите значение переменной s после выполнения фрагмента.

Бейсик	Python
<pre>n = 3 s = 0 FOR i = 0 TO 9 IF A(i) <= A(n) THEN t = A(i MOD n) A(i MOD n) = A(n) A(n) = t s = s + 1 END IF NEXT i</pre>	<pre>n = 3 s = 0 for i in range(0,10): if A[i] <= A[n]: t = A[i % n] A[i % n] = A[n] A[n] = t s = s + 1</pre>
Паскаль	Алгоритмический язык
<pre>n := 3; s := 0; for i:=0 to 9 do begin if A[i] <= A[n] then begin t := A[i mod n]; A[i mod n] := A[n]; A[n] := t; s := s + 1 end end;</pre>	<pre>n := 3 s := 0 нц для i от 0 до 9 если A[i] <= A[n] то t := A[mod(i,n)] A[mod(i,n)] := A[n] A[n] := t s := s + 1 все кц</pre>
C++	
<pre>n = 3; s = 0; for (i = 0; i <= 9; ++i) { if (A[i] <= A[n]) { t = A[i % n]; A[i % n] = A[n]; A[n] = t; s = s + 1; } }</pre>	

20. Ниже на пяти языках программирования записан алгоритм. Получив на вход число x , этот алгоритм печатает число M . Известно, что $x > 100$. Укажите наименьшее такое (т. е. большее 100) число x , при вводе которого алгоритм печатает 30.

Бейсик	Python
<pre>DIM X, L, M AS INTEGER INPUT X L = 2*X-30</pre>	<pre>x = int(input()) L = 2*x-30 M = 2*x+30 while L != M:</pre>

<pre> M = 2*X+30 WHILE L <> M IF L > M THEN L = L - M ELSE M = M - L END IF WEND PRINT M </pre>	<pre> if L > M: L = L - M else: M = M - L print(M) </pre>
Паскаль	Алгоритмический язык
<pre> var x, L, M: integer; begin readln(x); L := 2*x-30; M := 2*x+30; while L <> M do begin if L > M then L := L - M else M := M - L; end; writeln(M); end. </pre>	<pre> алг нач цел x, L, M ввод x L := 2*x-30 M := 2*x+30 нц пока L <> M если L > M то L := L - M иначе M := M - L все кц вывод M кон </pre>
Си++	
<pre> #include <iostream> using namespace std; int main() { int x, L, M; cin >> x; L = 2*x-30; M = 2*x+30; while (L != M) { if (L > M) L = L - M; else M = M - L; } cout << M << endl; return 0; } </pre>	

21. Определите, какое число будет напечатано в результате выполнения следующего алгоритма:

Бейсик	Паскаль
---------------	----------------

<pre> DIM A, B, T, M, R AS INTEGER A = -5: B = 5 M = A: R = F(A) FOR T = A TO B IF F(T) > R THEN M = T R = F(T) END IF NEXT T PRINT R FUNCTION F(x) F = (x+5)*(x+3) END FUNCTION </pre>	<pre> var a,b,t,M,R: integer; Function F(x:integer): integer; begin F := (x+5)*(x+3) end; begin a := -5; b := 5; M := a; R := F(a); for t := a to b do begin if (F(t) > R) then begin M := t; R := F(t) end end; write(R) end. </pre>
Си++	Алгоритмический
<pre> #include <iostream> using namespace std; int F(int x) { return (x+5)*(x+3); } int main() { int a, b, t, M, R; a = -5; b = 5; M = a; R = F(a); for (t = a; t <= b; t++) { if (F(t) > R) { M = t; R = F(t); } } cout << R << endl; } </pre>	<pre> алг нач цел а, b, t, M, R а := -5; b := 5 M := а; R := F(а) нц для t от а до b если F(t) > R то M := t; R := F(t) все кц вывод R кон алг цел F(цел x) нач знач := (x+5)*(x+3) кон </pre>
Python	
<pre> def f(x): return (x+5)*(x+3) a = -5 b = 5 M = a R = f(a) for t in range(a, b+1): if (f(t) > R): M = t R = f(t); print(R) </pre>	

22. Исполнитель РазДваТри преобразует число на экране.

У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1

2. Умножить на 2
3. Умножить на 3

Первая команда увеличивает число на экране на 1, вторая умножает его на 2, третья умножает его на 3.

Программа для исполнителя РазДваТри — это последовательность команд.

Сколько существует программ, которые преобразуют исходное число 2 в число 44 и при этом траектория вычислений содержит число 13 и не содержит числа 29?

Траектория вычислений — это последовательность результатов выполнения всех команд программы. Например, для программы 312 при исходном числе 6 траектория будет состоять из чисел 18, 19, 38.

23. 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?

$$(y_1 \rightarrow (y_2 \wedge x_1)) \wedge (x_1 \rightarrow x_2) = 1$$

$$(y_2 \rightarrow (y_3 \wedge x_2)) \wedge (x_2 \rightarrow x_3) = 1$$

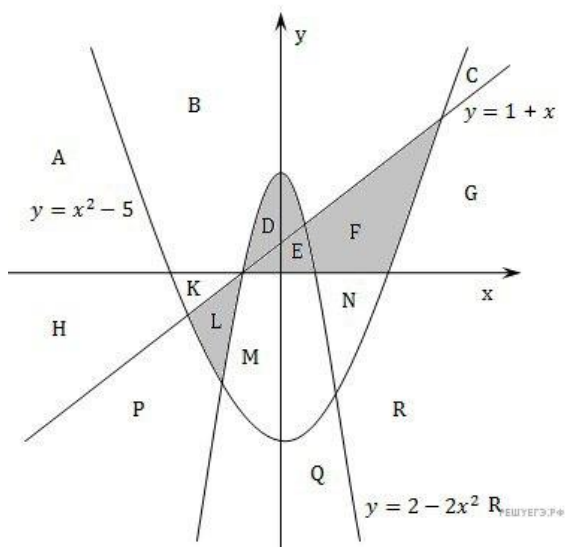
...

$$(y_6 \rightarrow (y_7 \wedge x_6)) \wedge (x_6 \rightarrow x_7) = 1$$

$$y_7 \rightarrow x_7 = 1$$

В ответе не нужно перечислять все различные наборы значений переменных $x_1, x_2, \dots, x_7, y_1, y_2, \dots, y_7$, при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких наборов.

24. Требовалось написать программу, при выполнении которой с клавиатуры считываются координаты точки на плоскости (x, y — действительные числа) и определяется принадлежность этой точки заданной закрашенной области (включая границы).



Ученик написал такую программу:

Паскаль	Бейсик
<pre>var x, y: real; begin readln(x,y); if y >= x+1 then begin if y <= 2-2*x*x then write('принадлежит') end else if y >= x*x-5 then write('принадлежит') else write('не принадлежит') end.</pre>	<pre>INPUT x, y IF y >= x+1 THEN IF y <= 2-2*x*x THEN PRINT "принадлежит" ELSE IF y >= x*x-5 THEN PRINT "принадлежит" ELSE PRINT "не принадлежит" END IF END IF END</pre>

Си++	Алгоритмический язык
<pre>#include <iostream> using namespace std; int main(){ float x, y; cin >> x >> y; if (y >= x+1) { if (y <= 2-2*x*x) cout << "принадлежит"; } else if (y >= x*x-5) cout << "принадлежит"; else cout << "не принадлежит"; }</pre>	<pre>алг нач вещ x, y ввод x, y если y >= x+1 то если y <= 2-2*x*x то вывод 'принадлежит' все иначе если y >= x*x-5 то вывод 'принадлежит' иначе вывод 'не принадлежит' все все кон</pre>
Python	
<pre>x = float(input()) y = float(input()) if y >= x+1: if y <= 2-2*x*x: print("принадлежит") else: if y >= x*x-5: print("принадлежит") else: print("не принадлежит")</pre>	

При проверке работы программа выполнялась по шагам для некоторых контрольных значений x и y , при этом был заполнен протокол тестирования. В результате неаккуратного обращения протокол был испорчен, частично сохранились только четыре строки:

Область	Условие 1 ($y \geq x+1$)	Условие 2 ($y \leq 2-2*x*x$)	Условие 3 ($y \geq x*x-5$)	Вывод	Верно
				—	
				принадлежит	нет
				не принадлежит	да
			—		да

Графы протокола содержат следующую информацию.

Область - часть плоскости, которой принадлежит проверяемая точка. (Все возможные области отмечены на рисунке буквами А, В, С, ... S.)

Условие 1, Условие2, Условие 3 — результат проверки соответствующего условия (да или нет). Если условие не проверялось, в протокол записывался прочерк.

Вывод — сообщение, которое вывела программа. Если программа ничего не вывела, в протокол записывался прочерк.

Верно - итоговое заключение (да или нет) о правильности результата работы программы при данных значениях x и y .

Последовательно выполните следующее.

1. Восстановите уцелевшие строки протокола, заполнив все клетки таблицы. Там, где содержание восстанавливается неоднозначно, запишите любое возможное значение. Например, если для нескольких областей получается одинаковая строка таблицы, укажите в графе «Область» любую из этих областей.

2. Укажите, как нужно доработать программу, чтобы не было случаев её неправильной работы. (Это можно сделать несколькими способами, достаточно указать любой способ доработки исходной программы.)

25. Дан массив, содержащий 2016 целых чисел. Необходимо найти и вывести сумму тех элементов этого массива, чётность которых совпадает с чётностью максимального элемента.

Например, в массиве из 6 элементов, равных соответственно 2, 3, 1, 5, 6, 4, максимальный элемент чётный (6), значит, ответом будет сумма чётных элементов этого массива $2 + 6 + 4 = 12$.

Напишите на одном из языков программирования программу для решения этой задачи. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных переменных.

Бейсик	Python
<pre>CONST N=2016 DIM A(N) AS INTEGER DIM I, M, S, P AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>	<pre># допускается также использо- # вание целочисленной # переменной m, s, p a = [] N = 2016 for i in range(0, N): a.append(int(input())) ...</pre>
Паскаль	Алгоритмический язык
<pre>const N=2016; var a: array [1..N] of integer; i, m, s, p: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>алг нач цел N=2016 целтаб a[1:N] цел i, m, s, p нц для i от 1 до N ввод a[i] кц ... кон</pre>
Си++	
<pre>#include <iostream></pre>	

```
using namespace std;
#define N 2016
int main(){
    int a[N];
    int i, m, s, p;
    for (i=0; i < N; i++)
        cin >> a[i];
    ...
    return 0;
}
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например Free Pascal 2.4). В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии.

26. Два игрока, Паша и Валя, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Паша. За один ход игрок может добавить в кучу 1 камень или удвоить количество камней в куче. Например, имея кучу из 7 камней, за один ход можно получить кучу из 8 или 14 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней. Игра завершается в тот момент, когда количество камней в куче S становится не менее 22. Победителем считается игрок, сделавший последний ход, если в куче осталось не менее 22 камней, но не больше 34 камней. Если же после завершающего хода игрока в куче оказывается больше 34 камней, то игрок, сделавший последний ход — проигрывает.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит, описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.

Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

- 1) а) При каких значениях Паша выигрывает 1 ходом. б) Кто выигрывает при $S=20, 19, 18$.
- 2) Кто выигрывает при $S=10, 9$.
- 3) Кто выигрывает при $S=8$. Нарисуйте дерево партий.

27. Вам предлагается два задания с похожими условиями: задание А и задание Б. Вы можете решать оба задания или одно из них по своему выбору. Задание Б более сложное, его решение оценивается выше. Итоговая оценка выставляется как максимальная из оценок за задания А и Б.

Задание А. Имеется набор данных, состоящий из 6 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. Если получить требуемую сумму невозможно, в качестве ответа нужно выдать 0.

Напишите программу для решения этой задачи. В этом варианте задания оценивается только правильность программы, время работы и размер использованной памяти не имеют значения.

Максимальная оценка за правильную программу – 2 балла.

Задание Б. 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. Если получить требуемую сумму невозможно, в качестве ответа нужно выдать 0.

Напишите программу для решения этой задачи.

Постарайтесь сделать программу эффективной по времени и используемой памяти (или хотя бы по одной из этих характеристик).

Программа считается эффективной по времени, если время работы программы пропорционально количеству пар чисел N , т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз.

Программа считается эффективной по памяти, если размер памяти, использованной в программе для хранения данных, не зависит от числа N и не превышает 1 килобайта.

Максимальная оценка за правильную программу, эффективную по времени и памяти, — 4 балла.

Максимальная оценка за правильную программу, эффективную по времени, но неэффективную по памяти, — 3 балла.

Как в варианте А, так и в варианте Б программа должна напечатать одно число — максимально возможную сумму, соответствующую условиям задачи (или 0, если такую сумму получить нельзя).

НАПОМИНАЕМ! Не забудьте указать, к какому заданию относится каждая из представленных Вами программ.

Перед текстом программы кратко опишите Ваш алгоритм решения, укажите использованный язык программирования и его версию (например, Free Pascal 2.6.4).

Входные данные

Для варианта А на вход программе подаётся шесть строк, каждая из которых содержит два натуральных числа, не превышающих 10 000.

Пример входных данных для варианта А:

```
1 3
5 12
6 9
5 4
3 3
1 1
```

Для варианта Б на вход программе в первой строке подаётся количество пар N ($1 \leq N \leq 100\,000$). Каждая из следующих N строк содержит два натуральных числа, не превышающих 10 000.

Пример входных данных для варианта Б:

```
6
1 3
5 12
6 9
5 4
3 3
1 1
```

Пример выходных данных для приведённых выше примеров входных данных: 32

Ключ

№ п/п	Ответ
1	30
2	yxzw
3	11
4	613
5	23

6	3
7	6
8	100
9	2
10	781
11	1392613
12	29
13	18
14	222111
15	40
16	4
17	20
18	12
19	3
20	120
21	80
22	150
23	36