

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**



**ОЛИМПИАДА ШКОЛЬНИКОВ
«ГРАНИТ НАУКИ»**

ИНФОРМАТИКА

**Методические указания для подготовки к
заключительному туру олимпиады в 2023/2024
учебном году
для 8 класса**

Санкт-Петербург
2024



ОГЛАВЛЕНИЕ

Информатика	3
Задания для подготовки к олимпиаде.....	6
Список рекомендуемой литературы для подготовки	21



Информатика

Характер и уровень сложности олимпиадных задач направлены на достижение целей проведения олимпиады: выявить способных участников, твердо владеющих школьной программой и наиболее подготовленных к освоению образовательных программ технических ВУЗов, обладающих логикой и творческим характером мышления, умеющих алгоритмически описать реальные ситуации из различных предметных областей и применить к ним наиболее подходящие методы информатики.

Задания дифференцированы по сложности и требуют различных временных затрат на верное и полное решение. Они охватывают все разделы школьной программы, но носят, в большинстве, комплексный характер, позволяющий варьировать оценки в зависимости от проявленных в решении творческих подходов и продемонстрированных технических навыков. Участники должны самостоятельно разбить задачу на подзадачи, грамотно выполнить решение каждой подзадачи, синтезировать решение всей задачи из решений отдельных подзадач. Успешное выполнение олимпиадной работы не предполагает знаний, выходящих за пределы школьной программы, но требует творческого подхода, логического мышления, умения увидеть и составить правильный и оптимальный план решения, четкого и технически грамотного выполнения каждой части решения.

Очный тур олимпиады проводится только в письменной форме. Каждый участник олимпиады получает билет, содержащий шесть заданий. При выполнении заданий требуется дать теоретическое обоснование и привести все этапы решения задачи:

- при определении результата перевода заданного числа из одной системы счисления в другую следует указать правило перевода в общем виде, а затем привести схему перевода заданного числа в указанную систему счисления;
- при ответе на вопрос об объемах памяти, выделяемой для кодирования символьной информации, необходимо указать особенности системы кодирования, привести подробные расчеты;
- при ответе на вопрос о поисковых запросах следует привести результаты промежуточных расчетов, полученных после выполнения запросов;



– при ответе на вопрос об определении значения переменной после выполнения программы, алгоритм которой приводится в билете, следует пояснить работу каждого оператора (блока), выписать промежуточные и окончательный результаты;

– необходимо дать развернутый ответ с пояснениями при составлении расширенного алгоритма, пояснить алгоритм графически;

– ответы на задачи, посвященные темам, связанным с сетями, IP-адресами, масками подсети должны сопровождаться публикацией всей информации без сокращений;

– в случае решения задач, связанных с таблицами истинности необходимо описывать все промежуточные результаты;

– при решении задачи на программирование необходимо составить блок-схему или описать алгоритм решения задачи, написать текст программы на любом из языков программирования, указав используемый язык программирования и его версию, допускается использовать только стандартную библиотеку языка;

– алгоритм должен раскрывать решение на каждом шаге. Код программы должен состоять из простых и понятных команд. Запрещено использовать уникальные особенности языка программирования (классы, методы, объекты, попарный обмен между переменными, специальные функции и библиотеки и др.) для реализации решения. Считывание заранее рассчитанных значений не засчитывается за решение задачи. Код должен содержать достаточное для понимания логики работы количество комментариев;

– программы, составленные участниками, тестируются и всесторонне оцениваются с учетом быстродействия, рациональности, минимального использования памяти компьютера и оригинальности предложенных решений.

При подготовке к олимпиаде следует повторить приведенные ниже разделы.

1. Информация и информационные процессы

1.1. Информация и ее кодирование

1.2. Виды информационных процессов

1.3. Процесс передачи информации, источник и приемник информации.

Сигнал, кодирование и декодирование.

1.4. Искажение информации

1.5. Дискретное (цифровое) представление текстовой, графической, звуковой информации и видеоинформации. Единицы измерения



количества информации

1.6.

1.7. Системы, компоненты, состояние и взаимодействие компонентов. Информационное взаимодействие в системе, управление, обратная связь

1.8. Моделирование

1.9. Математические модели

1.10. Системы счисления

1.11. Позиционные системы счисления

1.12. Двоичное представление информации

1.13. Логика и алгоритмы

1.14. Высказывания, логические операции, кванторы, истинность высказывания

1.15. Цепочки (конечные последовательности), деревья, списки, графы, матрицы (массивы), псевдослучайные последовательности

1.16. Вычислимые функции, полнота формализации понятия вычислимости, универсальная вычислимая функция

1.17. Кодирование с исправлением ошибок

1.18. Сортировка

1.19. Элементы теории алгоритмов

1.20. Формализация понятия алгоритма

1.21. Вычислимость. Эквивалентность алгоритмических моделей

1.22. Построение алгоритмов и практические вычисления

1.23. Языки программирования

1.24. Типы данных

1.25. Основные конструкции языка программирования. Система программирования

2. Средства ИКТ

3.1. Программная и аппаратная организация компьютеров и компьютерных систем. Виды программного обеспечения

3.5. Технологии создания и обработки текстовой информации

3.6. Понятие о настольных издательских системах. Создание компьютерных публикаций

3.7. Использование специализированных средств редактирования математических текстов и графического представления математических объектов

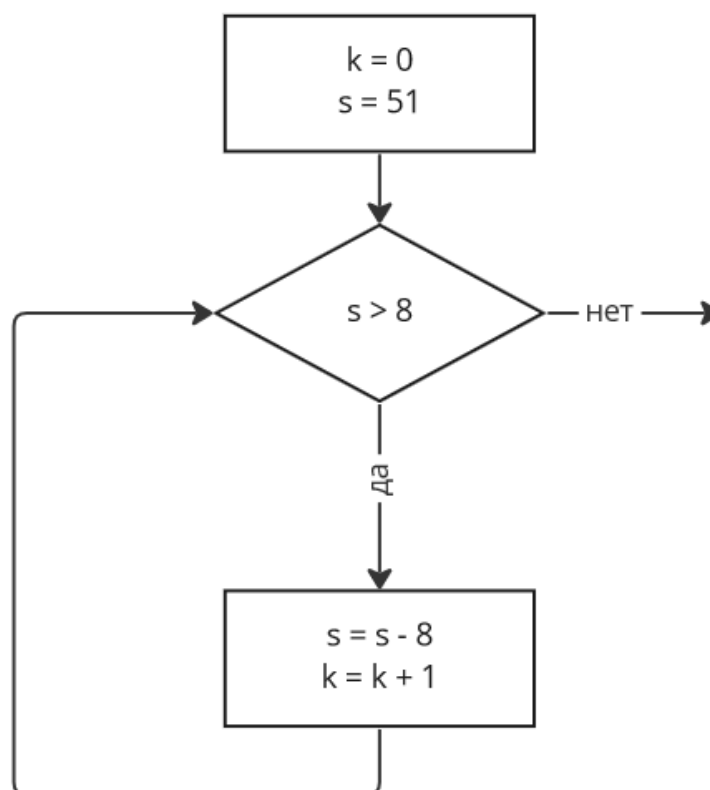
3.8. Технология создания и обработки графической и мультимедийной информации



- 3.9. Форматы графических и звуковых объектов
- 3.10. Ввод и обработка графических объектов
- 3.11. Ввод и обработка звуковых объектов
- 3.12. Обработка числовой информации
- 3.13. Технологии поиска и хранения информации
- 3.14. Использование инструментов поисковых систем (формирование запросов)
- 3.15. Телекоммуникационные технологии
- 3.16. Инструменты создания информационных объектов для Интернет

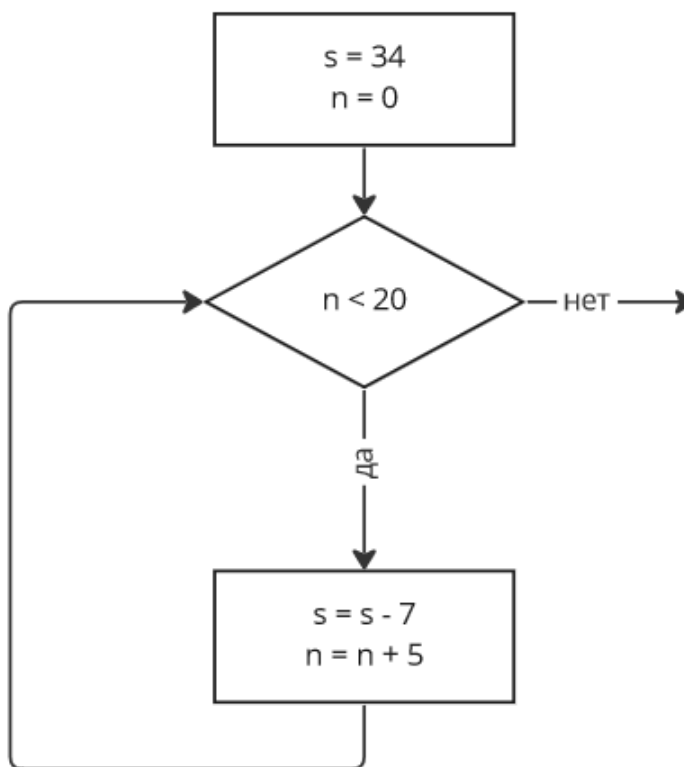
Задания для подготовки к олимпиаде

Задача 1. Дана блок-схема, чему равно «k» после выполнения алгоритма?



Решение: блок-схема сформирована с учетом условного оператора (ромба) с условием, что если условие выполнится, то будут произведены действия в блоке формул. Пока условие не станет ложным, блок будет повторяться. В итоге получится 6.
Ответ: (6)

Задача 2. Постройте блок-схему, если известно, что на вход поданы значения $s=34$, $n=0$, если $n < 20$, то, необходимо вычесть из s 7 и добавить к n 5. Повторять действия, пока условие не станет ложным. Укажите, чему будет равно s .

**Решение:**

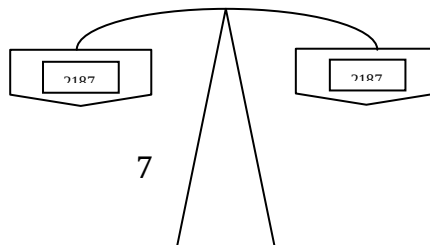
6.

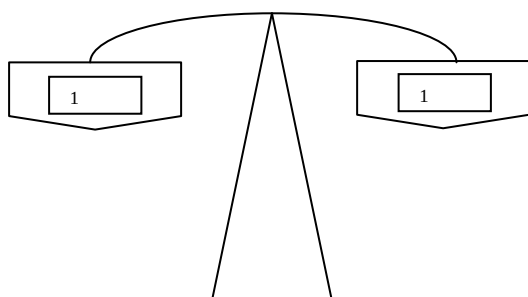
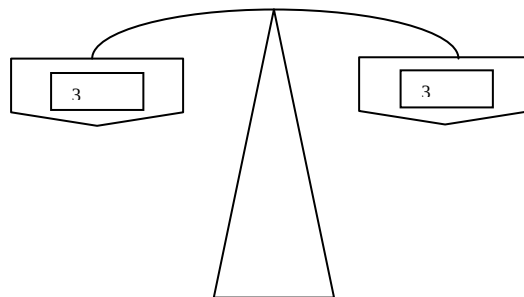
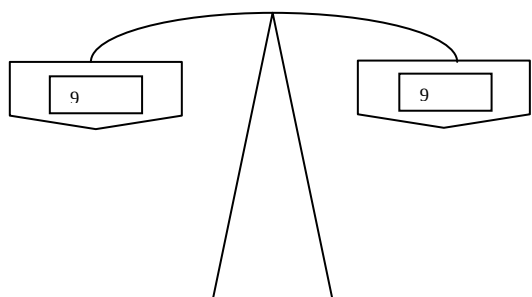
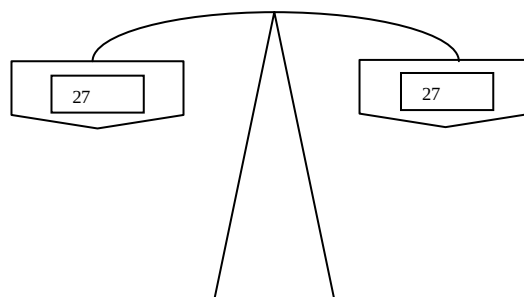
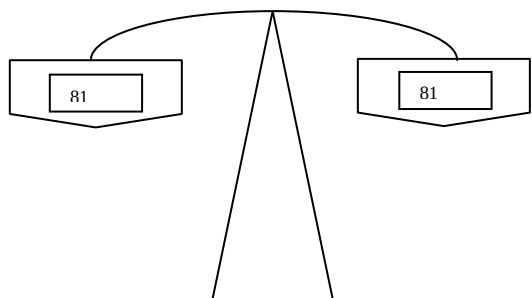
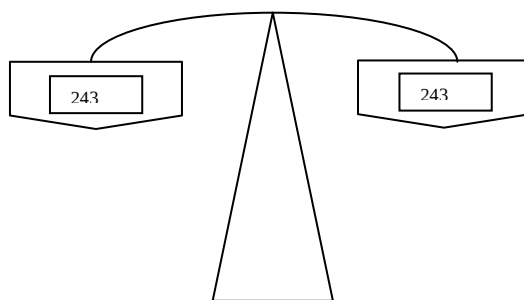
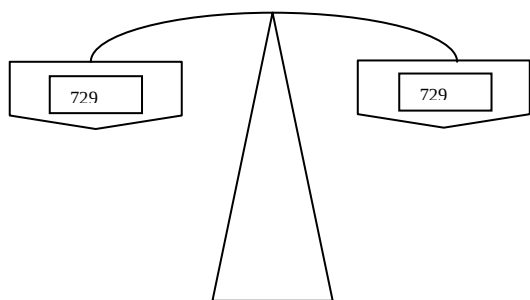
Задача 3. Гном держит в руках 6561 золотые монеты. Он точно знает, что одна монета фальшивая. Известно, что фальшивая монета весит меньше. У гнома есть весы с двумя чашами. С их помощью он планирует найти фальшивую монету, заполняя чаши равным количеством монет и ожидая, что весы придут в равновесие при отсутствии фальшивой монеты. Одну чашу весов можно заполнять любым количеством монет. За какое минимальное количество взвешиваний гному удастся это сделать? Поясните алгоритм расчёта количества взвешиваний графически. Какому критерию должно отвечать n – количество монет, чтобы подходить под тот же алгоритм взвешивания? Составьте универсальный алгоритм расчёта минимального количества взвешиваний для n монет, отвечающих сформулированному вами критерию.

Решение

Минимальное количество взвешиваний 9. Число монет должно быть кратно трём. Монеты разбиваются на три части. Две части взвешиваются. Если какая то из частей весит меньше, то фальшивая монета в этой части. Если 2 взвешиваемых части весят одинаково, то фальшивая монета в не взвешиваемой части. Далее опять разбиваем на три части и так пока не получим три части с 1 монетой. Какая легче и есть фальшивая.

Графический алгоритм:







Алгоритм:

Начало

- 1) Разделить монеты на три части.
- 2) Присвоить каждой части монет имена А, В, С.
- 3) Поместить части А и В на чаши весов.
- 4) Если $A < B$ то:

Выбираем часть А

Если $B < A$

Выбираем часть В

Иначе

Выбираем часть С

- 5) Если количество в выбранной части больше 1

Возвращаемся в шаг 1.

Иначе

Монета в выбранной части фальшивая.

Конец

Задача 3

Выражение $(x \rightarrow y) \vee y \wedge (\overline{z \rightarrow y}) \wedge \overline{x} = \text{ложь}$ Перечислите выражения, удовлетворяющие этому ответу из приведенных ниже:

- 1). $x \vee \overline{y} \wedge z \rightarrow x$
- 2). $\overline{y} \wedge x \leftrightarrow z \wedge x$
- 3). $\overline{x} \rightarrow y \wedge z \vee \overline{x}$
- 4). $z \leftrightarrow \overline{y} \vee \overline{x} \wedge z$
- 5). $x \rightarrow y \wedge x \wedge z$

Ответ: 2, 5.

Решение:

В качестве первого шага необходимо построить таблицу истинности:

x	y	z	$x \rightarrow y$	$z \rightarrow y$	$\overline{z \rightarrow y}$	$y \wedge \overline{z \rightarrow y}$	$\overline{x}$	$y \wedge \overline{z \rightarrow y} \wedge \overline{x}$	$(x \rightarrow y) \vee y \wedge (\overline{z \rightarrow y}) \wedge \overline{x}$
0	0	0	1	1	0	0	1	0	1
0	0	1	1	0	1	0	1	0	1
0	1	0	1	1	0	0	1	0	1
0	1	1	1	1	0	0	1	0	1
1	0	0	0	1	0	0	0	0	0
1	0	1	0	0	1	0	0	0	0
1	1	0	1	1	0	0	0	0	1
1	1	1	1	1	0	0	0	0	1



Ложное значение выражение принимает при значениях:

x	y	z
1	0	0
1	0	1

Необходимо проверить, какие из указанных выше выражений принимают ложное значение при заданных x, y, z . Для этого надо построить 5 таблиц истинности:

x	y	z	$\bar{y}$	$\bar{y} \wedge z$	$x \vee \bar{y} \wedge z$	$x \vee \bar{y} \wedge z \rightarrow x$
1	0	0	1	0	1	1
1	0	1	1	1	1	1

Данный комплект значений равен значению истина для первого выражения, следовательно, первое выражение не подходит. Проверим второе:

x	y	z	$\bar{y}$	$\bar{y} \wedge x$	$\bar{x}$	$z \wedge \bar{x}$	$\bar{y} \wedge x \leftrightarrow z \wedge \bar{x}$
1	0	0	1	1	0	0	0
1	0	1	1	1	0	0	0

Выражение №2 при заданных x, y, z принимает значение ложь, значит исходные подходят. Теперь проверяется третье выражение:

x	y	z	$\bar{x}$	$y \wedge z$	$y \wedge z \vee \bar{x}$	$z \wedge \bar{x}$	$\bar{x} \rightarrow y \wedge z \vee \bar{x}$
1	0	0	0	0	0	0	1
1	0	1	0	0	0	0	1

Судя по конечному результату (истина), данный комплект не подходит для выражения 3. Теперь необходимо проверить выражение №4:

x	y	z	$\bar{y}$	$\bar{x}$	$\bar{x} \wedge z$	$\bar{y} \vee \bar{x} \wedge z$	$z \leftrightarrow \bar{y} \vee \bar{x} \wedge z$
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	1

В результате данного комплекта значений получены значения ложь и истина, выражение не подходит под указанные значения. Осталось проверить последнее выражение:

x	y	z	$y \wedge x$	$\wedge x \wedge z$	$x \rightarrow y \wedge x \wedge z$
1	0	0	0	0	0
1	0	1	0	0	0

Данный комплект значений при подстановке в выражение дал значение ложь, поэтому также является подходящим.

Ответ: 2, 5.

Задача 4.

Катя получила в ответе 11.03 – это дата ее рождения. Ответ она получила, находя значения логического выражения. Укажите, какое логическое выражение надо было решить, чтобы получить такой ответ:

1. $A \& B \rightarrow C \wedge A$

2. $\bar{A} \& B \leftrightarrow C \wedge A$

3. $\bar{A} \& B \leftrightarrow \bar{C} \rightarrow \bar{A}$

Решение:

Расшифровав ответ, получаем, что в итоговом столбце должно быть 1103 =



01010011 – результирующий столбец таблицы истинности.

Для первого выражения этот столбец равен 11000101, переводя в десятичную систему каждые две цифры получится 3011.

Для второго выражения результирующий столбец равен 00001001, что будет равно 0021 в десятичной системе.

Третье выражение в результате дает 01010011 – 1103. Что подходит под указанное условие.

Задача 5.

Федор пришел на День Рождения к своему другу Илье, которому исполнилось 34 года. Федору 22 года, а вместе им 45 лет. Известно, что возраст Ильи и суммарное количество лет указано в одной системе счисления. Определите сколько лет каждому в десятичной системе счисления.

Решение:

Решить задачу можно, рассмотрев уравнение $34_x + 22_y = 45_y$

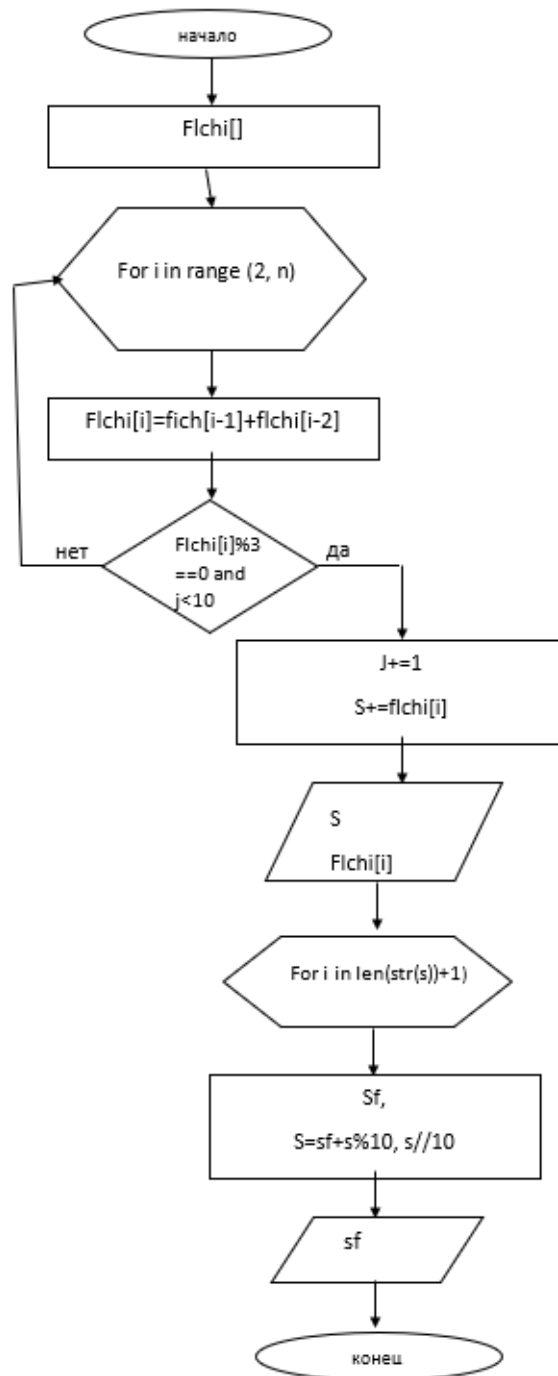
Из уравнения можно найти, чему равно в указанной системе счисления первое слагаемое: $45_y - 22_y = 23_y$. Получается, что $34_x = 23_y$. Стоит отметить, что все указанные числа не содержат в составе цифр больших 5. Отсюда следует предположение, что 34 – записано в пятиричной системе. В таком случае, $34_5 = 19_{10}$. Исходя из получившегося ранее равенства – $19_{10} = 23_x$, исходя из этого равенства, можно сделать вывод, что основание неизвестной системы счисления ненамного меньше, чем десятичная. Для начала рассмотрим перевод 19_{10} в число в девятиричной системе – 21, что не подходит. В восьмиричной системе – 23. Соответственно, равенство $23 + 22 = 45$ записано в системе счисления с основанием 8. Далее необходимо перевести $22_8 = x_{10}$. Искомое число – 18. Получается, что Илье 19 лет, а Федору 18.

Задача 6. Числа Фибоначчи – элементы числовой последовательности: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 610, 987, 1597, 2584, 4181, 6765, ..., в которой каждое последующее число равно сумме двух предыдущих чисел.

Среди чисел Фибоначчи встречаются числа, делящиеся на 3, 5 и 10. Определите сумму первых десяти чисел Фибоначчи, которые делятся на 3 и сумму цифр этих чисел.

Программа должна вывести первые 10 чисел Фибоначчи, делящиеся на 3, сумму этих чисел и сумму цифр этих чисел (не использовать встроенную или библиотечную функции вычисления чисел Фибоначчи).

Решение: в качестве решения сначала приведена блок-схема алгоритма:



Далее приведены примеры решения задачи на языке Python.

Решение с рекурсией (на Python):

```

def fichi(n): # Рекурсивная функция вычисления
    числа Фибоначчи
    if n < 2:
        return 1
    else:
        return fichi(n-1)+fichi(n-2)
  
```



```
n, i, s, sf = 0, 0, 0, 0
print('Первые 10 чисел, делящиеся на 3:')
while True:                                # организуем «вечный» цикл
    n += 1
    f = fichi(n)                            # вызов функции вычисления числа
    Фибоначчи
    if f % 3==0:                            # условие проверки делимости числа на
    3
        i += 1
        print(i, ' число =', f)            # вывод i-го числа
    Фибоначчи согласно условию
        s += f
        if i >=10:                         # условие выхода из «вечного»
        цикла: после вывода первых 10 чисел
            break                          # Фибоначчи, делящихся на 3
print('Сумма первых 10 чисел: s =', s)
for i in range(len(str(s))+1):
    sf, s = sf + s%10, s//10
print('Сумма цифр в этой сумме: sf =', sf)
```

Решение без рекурсии (на Python):

```
j, s, sf, n = 0, 0, 0, 1000                # 1000 – максимальное
        количество чисел Фибоначчи, задаем
fichi = [i for i in range(n)]
fichi[0], fichi[1] = 0, 1
print('Первые 10 чисел Фибоначчи, делящиеся на 3: ')
for i in range(2, n):
    fichi[i] = fichi[i-1] + fichi[i-2]
    if fichi[i]%3 == 0 and j<10:
        j +=1
        print(j, 'число =', fichi[i])
```



```
s += fichi[i]

print('Сумма первых 10 чисел: s =', s)
for i in range(len(str(s))+1):
    sf, s = sf + s%10, s//10
print('Сумма цифр в этой сумме: sf =', sf)
```

```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: D:\Олимпиада_шк_Гранит науки(2020в)\Фиб2020\fib_v3.py ====
Первые 10 чисел, делящиеся на 3:
1 число = 3
2 число = 21
3 число = 144
4 число = 987
5 число = 6765
6 число = 46368
7 число = 317811
8 число = 2178309
9 число = 14930352
10 число = 102334155
Сумма первых 10 чисел: s = 119814915
Сумма цифр в этой сумме: sf = 39
>>>|
```

Задача 7.

При движении муравьи образуют ряды так, что каждый муравей, кроме первого, находится позади другого муравья. Некоторые люди задаются вопросом, а что происходит, когда два ряда муравьев, движущихся в противоположных направлениях, сталкиваются друг с другом в проходе, слишком узком для обоих рядов. Одна из теорий гласит, что в такой ситуации муравьи будут перепрыгивают друг через друга.

С момента, когда ряды муравьев встречаются, каждую секунду каждый муравей перепрыгивает (или перебирается, так что оба муравья поменялись местами, но только если другой муравей движется в противоположном направлении).

Напишите программу, которая выводит порядок следования муравьев через T секунд.

Входные данные: первая строка содержит два целых числа $N1$ и $N2$ - количество муравьев в первом и втором рядах, соответственно.

Следующие две строки содержат порядковые номера муравьев в первом и втором ряду (от первого до последнего). Каждый муравей однозначно определяется заглавной буквой английского алфавита (эта буква уникальна для обоих строк). Последняя строка ввода содержит целое число T ($0 \leq T \leq 50$).

Результат должен содержать порядок муравьев через T секунд в одной строке. Наша точка зрения такова, что первый ряд муравьев идет с левой стороны, а второй - с правой.

Ввод	Результат
3 3 ABC DEF	CBADEF



0	
3 3 ABC DEF 2	CDBEAF
3 4 JLA CRUO 3	CARLUJO

Исходные данные:

14 12

ABCDEFGHIJKLMN

OPQRSTUVWXYZ

20

Ответ:

OPQRSTUNVMWLXKYJZIHGFEDCBA

Решение:

Один из подходов заключается в посекундном моделировании поведения муравьев.

Более эффективный подход напрямую вычисляет конечную позицию каждого муравья.

Давайте обозначим муравьев номерами в их рядах, начиная с 0

Мы знаем начальную ($T = 0$) позицию P каждого муравья: $P = N1 - i - i1$ для первого (левого) ряда и $P = N1 + I$ для второго (правого).

Есть три случая:

1. Если T меньше i , то муравей не будет перепрыгивать через других муравьев и останется на своей начальной позиции.
2. Если T больше или равно $i +$ (количество муравьев в другом ряду), то муравей перепрыгнет через всех муравьев в другом ряду.
3. В противном случае позиция муравья изменится на $(T - i)$.

Решение на C++

```
#define _CRT_SECURE_NO_WARNINGS
#include <algorithm>
#include <cstdio>

using namespace std;

int main()
{
    char k1[31], k2[31], res[61];
    int n1, n2, t, P, i;

    scanf("%d%d", &n1, &n2);
```



```
scanf("%s%s", k1, k2);
scanf("%d", &t);

for (i = 0; i < n1; i++)
{
    P = n1 - i - 1;
    P += min(max(t - i, 0), n2);
    res[P] = k1[i];
}
for (i = 0; i < n2; i++)
{
    P = n1 + i;
    P -= min(max(t - i, 0), n1);
    res[P] = k2[i];
}
res[n1 + n2] = '\\0';
printf("%s\\n", res);

return 0;
}
```

Решение на Python

```
n1, n2 = map(int, input().split())
k1 = input()
k2 = input()
t = int(input())

res = [''] * (n1 + n2)

for i in range(n1):
    P = n1 - i - 1
    P += min(max(t - i, 0), n2)
    res[P] = k1[i]

for i in range(n2):
    P = n1 + i
    P -= min(max(t - i, 0), n1)
    res[P] = k2[i]

print(''.join(res))
```


**Список рекомендуемой литературы для подготовки**

1. Босова Л.Л., Информатика. Учебник. Базовый уровень - М.: Просвещение. 2023 – 176 с.
2. Семакин И.Г., Залогова Л.А., Русаков С.В., Шестакова Л.В., Информатика. Базовый уровень: учебник для 11 класса. – 3-е изд. – Просвещение, 2021. – 176 с.
3. Босова Л.Л. Информатика. 7-9 классы. Компьютерный практикум. / Л.Л. Босова, А.Ю. Босова, Н.А. Аквилянов. – М.: Бином. Лаборатория знаний, 2021. – 192 с.
4. Залогова Л.А., Информатика и ИКТ. 8-11 класс. Задачник-практикум, Л.А. Залогова, И.Г. Семакин, Е.К. Хеннер, 2015 – 608 с.
5. Босова Л.Л. Информатика. 8-9 классы. Начала программирования на языке Python. Дополнительные главы. ФГОС, Босова Л.Л., Аквилянов Н.А., Кочергин И.О. – М.: «Просвещение». 2023 – 96 с.
6. Ушаков Д.М., ОГЭ-2024. Информатика. 10 вариантов. Типовые варианты экзаменационных заданий: Экзамен, 2024. – 120 с.
7. Зюзьков В. М. Математическая логика и теория алгоритмов : учебное пособие /В. М. Зюзьков. — Томск : Эль Контент, 2015. — 236 с.
8. А.А. Майоров, В.П. Седякин Общая и теоретическая информатика: учебное пособие. — М.: МИИГАиК, 2017.— 128 с.
9. Сборник олимпиадных задач по информатике : практикум. В 5 ч. Ч. 1/ С. И. Кашкевич, А. А. Толстиков. – Минск : БГУ, 2016. – 29 с.
10. Златопольский Д.М. Занимательная информатика. – М.: Бином.Лаборатория знаний, 2011. – 424 с.
11. Кирюхин В.М., Цветкова М.С. Информатика. Программы внеурочной деятельности учащихся по подготовке к Всероссийской олимпиаде школьников: 5–11 классы. – М.: БИНОМ. Лаборатория знаний, 2014. – 224 с.
12. Кудинов Ю. И., Пащенко Ф. Ф., Келина А. Ю. Практикум по основам современной информатики. – СПб. : Лань, 2011 . – 352 с.
13. Малярчук С. Н. Информатика в определениях, таблицах и схемах. – Харьков: «Ранок». 2011 – 112с.
14. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. 4-е изд. – СПб.: Питер, 2010. – 918 с.
15. Сафронов И.К. Готовимся к ЕГЭ. Информатика. – СПб.: ВНУ-Санкт-Петербург. 2009 – 368 с.
16. Р. Авданин, Программирование: вводный курс, 1995 – МЦМНО – 100с.
17. Триумфгородских М. Дискретная математика и математическая логика для информатиков, экономистов и менеджеров. – М.:«Диалог-МИФИ». 2011 – 180 с.

Составители: профессор каф.ИиКТ *Кризский В.Н.*, профессор каф.ИСВТ *Иванова И.В.*,
доцент каф. ИиКТ *Иванов И.В.*, доцент каф. ИиКТ *Кочнева А.А.*, ассистент каф. ИСВТ *Киба М.Р.*