

Тренировки по алгоритмам 5.0 от Яндекса — Занятие 1 (Сложность, тестирование, особые случаи)

10 мар 2024, 17:10:49
старт: 1 мар 2024, 20:30:00
финиш: 12 мар 2024, 18:00:00
до финиша: 2д.
начало: 1 мар 2024, 20:30:00
конец: 12 мар 2024, 18:00:00
длительность: 10д. 21ч.

I. Расписание

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Как же Илье надоело учиться! Сначала школа, потом университет... Вот, наконец, наступил тот долгожданный день, когда Илье утром не надо ехать на учебу. Но, к несчастью для Ильи, оказалось, что после окончания университета начинается самое трудное — надо устраиваться на работу.

Во всемирно известной фирме «Goondex», в которую устроился Илья, принято очень много работать, в частности, для сотрудников установлена шестидневная рабочая неделя. Но, в качестве бонуса, «Goondex» каждый год предлагает своим сотрудникам выбрать любой день недели в качестве выходного. В свою очередь, оставшиеся шесть дней недели будут рабочими.

Илья сообразил, что с учётом государственных праздников (которые всегда являются выходными) с помощью правильного выбора выходного дня недели можно варьировать количество рабочих дней в году. Теперь он хочет знать, какой день недели ему следует выбрать в качестве выходного, чтобы отдыхать как можно больше дней в году, или, наоборот, демонстрировать чудеса трудолюбия, работая по максимуму.

Формат ввода

В первой строке входных данных находится одно целое число N ($0 \leq N \leq 366$) — количество государственных праздников.

Во второй строке содержится одно целое число $year$ ($1800 \leq year \leq 2100$) — год, в который необходимо помочь Илье.

В каждой из последующих N строк расположено по паре чисел $day\ month$ (day — целое число, $month$ — слово, между day и $month$ ровно один пробел), обозначающих, что день day месяца $month$ является государственным праздником.

В последней строке расположено слово day_of_week — день недели первого января в год $year$.

Гарантируется, что все даты указаны корректно (в том числе указанный день недели первого января действительно является днём недели первого января соответствующего года $year$) и все дни государственных праздников различны.

Формат вывода

Выведите через пробел два дня недели — лучший и худший варианты дней недели для выходного (то есть дни недели, для которых достигается соответственно максимальное и минимальное количество выходных дней в году). Если возможных вариантов ответа несколько, выведите любой из них.

Пример 1

Ввод	<input type="text"/>	Вывод	<input type="text"/>
2		Monday	Thursday
2015			
1 January			

Ввод

Вывод

8 January
Thursday

Пример 2

Ввод

Вывод

3
2013
1 January
8 January
15 January
Tuesday

Monday Tuesday

Пример 3

Ввод

Вывод

3
2013
6 February
13 February
20 February
Tuesday

Tuesday Wednesday

Примечания

Рассмотрим подробно **третий пример**.
2013 год начинается и заканчивается во вторник (Tuesday), при этом на вторник приходится 53 дня года, а на все остальные дни недели — по 52 дня. Все три государственных праздника выпадают на среду (Wednesday). Если Илья выберет в качестве выходного дня вторник, то в году у него будет $53 + 3 = 56$ выходных дней (53 вторника и 3 государственных праздника). Если Илья выберет в качестве выходного дня среду, то у него будет только 52 выходных дня. Если же Илья выберет в качестве выходного дня любой другой день недели, то у него будет $52 + 3 = 55$ выходных дней.

Таким образом, лучший вариант для выходного дня — вторник, худший — среда, и **единственным** правильным ответом в данном примере является Tuesday Wednesday.

Соответствие названий месяцев и дней недели в английском и русском языках:

Месяцы				Дни недели	
January	Январь	July	Июль	Monday	Понедельник
February	Февраль	August	Август	Tuesday	Вторник
March	Март	September	Сентябрь	Wednesday	Среда
April	Апрель	October	Октябрь	Thursday	Четверг
May	Май	November	Ноябрь	Friday	Пятница
June	Июнь	December	Декабрь	Saturday	Суббота
				Sunday	Воскресенье

Соответствие названий месяцев и количества дней в них:

Январь	31	Июль	31
Февраль	28 (29)	Август	31
Март	31	Сентябрь	30
Апрель	30	Октябрь	31
Май	31	Ноябрь	30
Июнь	30	Декабрь	31

В феврале 29 дней только в високосные года. Год является високосным, если он кратен 400, либо кратен 4 и не кратен 100. Например, 1996 и 2000 являются високосными, а 1999 и 1900 — нет.

Язык

Python 3.12.1

Набрать здесь

Отправить файл

```
1 n = int(input())
2 year = int(input())
3
4 holidays = []
5
6 month = {'January': 31, 'February': 28, 'March': 31, 'April': 30, 'May': 31, 'June': 30, 'July': 31, 'August': 31, 'September':
7 monthv = {'January': 31, 'February': 29, 'March': 31, 'April': 30, 'May': 31, 'June': 30, 'July': 31, 'August': 31, 'September':
8
9 for _ in range(n):
10     d, m = map(str, input().split())
11     holidays.append((int(d), m))
12
13 start = str(input())
14
15 days_week = {'Monday': 0, 'Tuesday': 1, 'Wednesday': 2, 'Thursday': 3, 'Friday': 4, 'Saturday': 5, 'Sunday': 6}
16 days = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday']
17 days_cnt = {'Monday': 0, 'Tuesday': 0, 'Wednesday': 0, 'Thursday': 0, 'Friday': 0, 'Saturday': 0, 'Sunday': 0}
18
19
20 if year%400 == 0 or (year%4 == 0 and year%100 != 0):
21     weeks = 366//7
22
23     for i in days:
24         days_cnt[i] += weeks
25
26     start_week = days.index(start)
27     last_week = 366 - weeks * 7
28
29     for j in range(start_week, last_week + start_week):
30         if j < 7:
31             days_cnt[days[j]] += 1
32         else:
33             j -= 7
34             days_cnt[days[j]] += 1
35
36     for hday in holidays:
37         date = 0
38
```

Отправить

Предыдущая

Следующая