2016 網際網路程式設計全國大賽
國中組初賽

- 本次比賽共 6 題，含本封面共 14 頁。
- 全部題目的輸入都來自**標準輸入**。輸入中可能包含多組輸入，以題目敘述為主。
- 全部題目的輸出皆輸出到螢幕 (**標準輸出**)。
  輸出和裁判的答案必須完全一致，英文字母大小寫不同或有多餘字元皆視為答題錯誤。
- 所有題目的時間限制請參考 Kattis 網頁上各題之標示。
- 比賽中上傳之程式碼，使用 C 語言請用 `.c` 為副檔名；使用 C++ 語言則用 `.cpp` 為副檔名。
- 使用 `cin` 輸入速度遠慢於 `scanf` 輸入，若使用需自行承擔 Time Limit Exceeded 的風險。
- 部分題目有浮點數輸出，會採容許部分誤差的方式進行評測。一般來說「相對或絕對誤差小於 ϵ 皆視為正確」，ϵ 值以題目敘述為主。

  舉例來說，假設 $\epsilon = 10^{-6}$ 且 a 是正確答案，b 是你的答案，如果符合 $\frac{|a-b|}{\max(|a|,|b|,1)} \leq 10^{-6}$，就會被評測程式視為正確。

	題目名稱
題目 A	貓貓與甜甜圈
題目 B	貓貓與汽車
題目 C	保羅的寶貝
題目 D	普遜發糖果
題目 E	普遜的減肥計劃
題目 F	頗旺想放假

2016 網際網路程式設計全國大賽
輸入輸出範例

C 程式範例：

```c
#include <stdio.h>
int main()
{
    int cases;
    scanf("%d", &cases);
    for (int i = 0; i < cases; ++i)
    {
        long long a, b;
        scanf("%lld %lld", &a, &b);
        printf("%lld\n", a + b);
    }
    return 0;
}
```

C++ 程式範例：

```cpp
#include <iostream>
int main()
{
    int cases;
    std::cin >> cases;
    for (int i = 0; i < cases; ++i)
    {
        long long a, b;
        std::cin >> a >> b;
        std::cout << a + b << std::endl;
    }
    return 0;
}
```

A. 貓貓與甜甜圈

Problem ID: donut

普遜是一隻喜歡吃飯和睡覺的貓貓，牠最喜歡吃的東西就是甜甜圈了。

而有名的甜甜圈連鎖店，多拿滋嚼嚼 (Donut Chump) 要在普遜的住處附近開分店了，他們的口號是推廣甜甜圈的美好，讓甜甜圈再次偉大！

普遜聽到了這個消息後就興奮的跑到了多拿滋嚼嚼買了 N 個甜甜圈回家，每個甜甜圈的口味都不一樣。多拿滋嚼嚼的包裝方式是把所有的甜甜圈疊成一個高塔狀，也就是 N 個甜甜圈會被疊成 N 層，每層一個甜甜圈。而普遜吃甜甜圈的方法是從最上層開始，依序把所有的甜甜圈吃光光。

在普遜讀過了名為「先別急著吃甜甜圈」的一本書後，牠認為不應該一次把所有的甜甜圈吃完，而是應該要留一些甜甜圈之後再來吃。因此，普遜想出了一個叫做「嚼嚼吃法」的甜甜圈吃法：先把最上面的若干層的甜甜圈冰到冰箱，然後再開始由上往下依序吃甜甜圈，吃到某個時候停下來，然後把剩下的甜甜圈也冰進冰箱。不過吃甜甜圈前可以不把任何甜甜圈冰到冰箱，也可以開始吃甜甜圈之後就一口氣把剩下的甜甜圈都吃完，但至少要吃掉一個甜甜圈。也就是說，把所有甜甜圈全部吃光也算是一種嚼嚼吃法。

對普遜而言，每個口味的甜甜圈都有一個滿足度，一個吃法的滿足度就是所有被普遜吃掉的甜甜圈的滿足度的總和。在普遜檢視了自己的慾望後，決定要選所有嚼嚼吃法當中，滿足度第 K 大的吃法來享用這些甜甜圈。但普遜的腦袋不太好，因此牠想要請你寫一個程式，判斷在已經知道每個甜甜圈的滿足度的前提下，滿足度第 K 大的甜甜圈吃法的滿足度是多少呢？

例如，如果普遜買了三個甜甜圈，由上往下的滿足度依序為 $(1, 2, 3)$ ，那麼這些甜甜圈有 6 種嚼嚼吃法：$\{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 2, 3\}$，滿足度分別為 $1, 2, 3, 3, 5, 6$ ，其中滿足度第二大的嚼嚼吃法的滿足度是 5 ，滿足度第三大和第四大的嚼嚼吃法的滿足度都是 3 。

Input

測試資料第一行有兩個整數 N, K ，分別表示甜甜圈的數量及普遜想要找的嚼嚼吃法的滿足度次序。

測試資料第二行有 N 個正整數 a_i 表示從最上層數來的第 i 個甜甜圈的滿足度。

- $1 \leq N \leq 10^5$

- $1 \leq K \leq \min(\frac{N \times (N+1)}{2}, 10^9)$
- $1 \leq a_i \leq 10^4$

Output

輸出一個整數，表示滿足度第 K 大的嚼嚼吃法的滿足度是多少。

Sample Input 1	Sample Output 1
6 1 8 7 1 2 2 8	28

Sample Input 2	Sample Output 2
3 4 1 2 3	3

B. 貓貓與汽車

Problem ID: meow

有一個經典的數學問題是這樣的：

你被邀請來參加一個贈獎活動，眼前有 3 扇門，其中 1 扇門後面有一輛汽車、其餘 2 扇門後面各有一隻貓貓，你會被要求以一個特殊的方式選出一扇門，然後你會得到你最終選取的門後面的物品。

活動開始時，你先從 3 扇門當中選取其中一扇，接著負責主持這個活動的人會從沒有被你選取到的 2 扇門中開啟一扇，且那扇門後面一定是貓貓，之後給你一次機會選擇要不要更換你選取的門。然後活動結束，你就會得到最終選取的門後面的物品。

若你想要得到貓貓，你應不應該更換自己選取的門呢？

事實上這個問題的解答是你不該更換，至於為什麼呢... 那又是另一個故事了。

今天有個人覺得 3 扇門太不夠看了，因此他要舉辦同樣的活動，但是一開始眼前將會有 N 扇門。一樣只會有 1 扇門後面是汽車，其餘的 $N-1$ 扇門後面都會是貓貓；參與者先選取 1 扇門之後，主持人會從沒被選到的 $N-1$ 扇門中挑選 $N-2$ 扇後面是貓貓的門並開啟這些門。

然而問題來了，門的數量實在太多了，由一個人來挑選要開啟的門實在太困難了，因此要請你寫一個程式來選出要開啟的這 $N-2$ 扇門。

Input

測試資料第一行包含一個正整數 N，表示門的數量，門的編號依序為 1 到 N。

測試資料第二行包含一個長度為 N 的字串 S，字串 S 的第 i 個字元用來描述第 i 扇門後的物品，若為字元 `'C'` 表示此扇門後為汽車、若為字元 `'M'` 表示此扇門後為貓貓。

測試資料第三行包含一個正整數 K，表示參與者選取了編號為 K 的門。

- $3 \le N \le 514$
- 字串 S 恰由 1 個字元 `'C'` 跟 $N-1$ 個字元 `'M'` 組成
- $1 \le K \le N$

Output

輸出一行，包含 $N-2$ 個相異整數，表示選取要開啟的門的編號。每個數字之間要恰有一個空格，結尾記得要換行（但換行前不能有空格，所以整個輸出應該會包含 $N-3$ 個空格）。

若選取方法有多種，可以輸出任意一種。另外，也可以以任意順序輸出門的編號。

Sample Input 1	Sample Output 1
3 MMC 1	2

Sample Input 2	Sample Output 2
4 MMCM 2	4 1

C. 保羅的寶貝

Problem ID: badboy

保羅有 N 個寶貝放在桌子上，他想要把它們一個一個搬回櫃子裡好好保存。但是保羅的寶貝們都很重，保羅想要盡量輕鬆地完成搬運的工作。

而保羅有 M 個拿來保存寶貝的櫃子，其中每個櫃子最多只能裝一個寶貝，因為每個寶貝都要分別的保存才不會損壞。

保羅希望搬運的疲勞程度越小越好，所謂疲勞程度就是每個寶貝重量乘上搬運該寶貝的距離的總和。而且因為寶貝很脆弱，所以保羅每次只能搬一個寶貝，搬完就得要回到桌子搬下一個寶貝。此外，沒有搬運寶貝時任何走動都不會累積疲勞程度。

保羅已經知道 N 個寶貝的重量與 M 個存放寶貝的櫃子離桌子的距離，他想知道他搬運寶貝疲勞程度最小是多少？

Input

測試資料第一行有兩個正整數 N, M，分別表示寶貝與櫃子的數量。

測試資料第二行會有 N 個由空格隔開的正整數 $w_1, w_2, \ldots, w_N$，代表每個寶貝的重量。

測試資料第三行會有 M 個由空格隔開的正整數 $d_1, d_2, \ldots, d_M$，代表每個櫃子離桌子的距離。

- $1 \leq N \leq M \leq 10^6$
- $1 \leq$ 每個寶貝的重量 $\leq 10^4$
- $1 \leq$ 每個櫃子的距離 $\leq 10^4$

Output

請輸出一個正整數於一行，代表保羅的最小疲勞程度。

Sample Input 1	Sample Output 1
5 6 10 2 1 514 4 1 2 100 2 3 9	557

D. 普遜發糖果

Problem ID: candy

又到了普遜期待已久的萬聖節，今年他和班上同學一起喊著「不給糖，就搗蛋」的口號，走上街頭挨家挨戶收集糖果。由於普遜實在太可愛了，走了一大圈後，普遜和他的夥伴們收集了大量的糖果。但時間已經晚了，大家都該回家了，所以普遜得想個方法把糖果分給大家帶回去。這時普遜想到了一個簡單的分配方法：普遜班上共有 N 個人，每個人有個 1 到 N 的座號。依照大家在班上的座號，由 1 號開始，一人拿走一個糖果，N 號拿完後再輪回 1 號繼續拿，一直這樣進行直到糖果都被分完。

而在分配的過程中，好奇的普遜想在分配完之前搶先知道自己最後可以帶走幾個糖果，所以他把班上的人數 N ，總共收集的糖果數 M 以及普遜自己的座號 k 都告訴你了，希望聰明的你可以幫他算出正確的數量。

Input

測試資料第一行有一個正整數 T，代表接下來有 T 次分配的過程。

每次分配的過程包含一行，包含三個正整數 N, M, k，代表普遜班上共有 N 位同學，普遜是其中的 k 號，且這次萬聖節他們一起收集了 M 個糖果。

- $1 \leq T \leq 500$
- $20 \leq N \leq 100$
- $N \leq M \leq 10000$
- $1 \leq k \leq N$

Output

對於每次分配，請輸出一行包含一個正整數代表普遜最後會被分配到的糖果數量。

Sample Input 1	Sample Output 1
3 25 100 2 25 102 2 25 102 3	4 5 4

E. 普遜的減肥計劃

Problem ID: diet

普遜是一隻可愛的貓貓，牠平時的興趣是吃東西還有睡覺。

經過了漫長而快樂的暑假，普遜發現自己的體重居然上升了 7.122 公斤 !! 普遜覺得如果再這樣下去，有一天自己會因為肚子太圓，手跟嘴巴都碰不到食物而再也無法進食，為了避免這種事發生，普遜決定要開始減肥。

減肥的原則就是少吃和多動，但普遜是隻討厭運動的貓貓，牠跳到紙箱裡想了很久後，終於想到，最好的運動應該還是睡覺吧，很多時候睡覺起來的時候都會感覺很累，想必睡覺的運動量其實挺大的嘛！

說時遲那時快，普遜一不小心就排出了行程表，表上以一小時為單位，上面寫了這一個小時要拿來睡覺還是要拿來吃東西，而且行程表上睡覺的總時數與吃東西的總時數一樣多。但是普遜有一個特別的想法，如果在減肥計畫開始之後，有任何一個瞬間，普遜累計花在吃東西上的時間比牠累計花在睡覺上的時間還要多，他就會認為自己的減肥計劃失敗了，然後難過地躺在地上玩毛線球。

為了維護世界的和平，普遜想要更動自己的行程表，以確保自己的減肥計劃可以成功。然而，重新規劃整個行程表實在是太麻煩了，於是普遜想到了一個做法：把行程表剪成兩半，然後把最前面的一部分移到行程表的最後。例如：原本的行程表是「睡, 吃, 吃, 睡, 吃, 睡」，那麼在第三個行程結束後，吃的時間就比睡的時間多了，為了避免這件事發生，普遜可以把前三個行程，也就是「睡, 吃, 吃」移到最後，讓行程變為「睡, 吃, 睡, 睡, 吃, 吃」，這樣過程中就沒有任何一個瞬間，花在吃東西上面的時間比花在睡覺的時間多了！

但這個問題實在太困難了，所以普遜想請你寫個程式，給你牠的行程表，告訴普遜應該要最前面的多少個行程移到最後面，才可以避免減肥計劃失敗？如果有多個方法的話，請輸出搬動最少個行程的方法。

Input

測試資料第一行有一個整數 N ，表示普遜行程表上行程的數量。

測試資料第二行會有一個由 `'s'` 和 `'e'` 組成的字串 S ，s_i 若為 `'s'` 表示第 i 個行程是睡覺，反之表示第 i 個行程是吃東西。

- $2 \leq N \leq 10^5$, N 為偶數
- $s_i \in \{$`'s'`, `'e'`$\}$

Output

請輸出一行，表示普遜應該要把幾個行程搬到最後面。如果沒有解的話請輸出一行 "-1" (不包含引號)。

Sample Input 1	Sample Output 1
6 seeses	3

Sample Input 2	Sample Output 2
2 se	0

F. 頗旺想放假

Problem ID: happy

頗旺是地虎國一個很認真的學生，每個禮拜星期一到星期五都會到學校上課，而星期六及星期天便可以在家裏休息，做其他有趣的課餘活動（如參加 NPSC）。

而地虎國是一個潮溼的熱帶小島，因此，時不時便會有颱風來肆虐。只要颱風來襲，爲了保護所有人民的安危，全國便會放假一天。

而對頗旺而言，如果在星期一到星期五的時候，恰好有颱風來襲，便會賺到一天在家裏休息的機會，頗旺便會覺得很開心；反之，若沒有颱風，便一樣要到學校上課，頗旺便會覺得不開心。但如果是在星期六及星期日時，有颱風來襲，原本可以好好放假的心情，便會被外面的大風大雨給影響，頗旺便會覺得不開心：反之，若沒有颱風，頗旺便會覺得很開心。

現在，你剛好發現一頁頗旺的日記，上面記著當天的年份、月份、日期、星期幾、以及有沒有颱風，你很好奇頗旺當天是開心還是不開心。

Input

測試資料只有一行包含五個整數 Y, M, D, X, T，分別代表年份、月份、日期、星期幾、有沒有颱風。當 $X = 7$，代表當天是星期天。當 $T = 0$ 代表當天沒有颱風、$T = 1$ 代表當天有颱風。

- $1 \leq Y \leq 9999$
- $1 \leq M \leq 12$
- $1 \leq D \leq 31$
- $1 \leq X \leq 7$
- $0 \leq T \leq 1$
- 保證 Y 年 M 月 D 日存在

Output

如果頗旺當天開心，請輸出 "happy" 於一行，否則請輸出 "unhappy" 於一行，皆不包含引號。輸出後請記得換行。

Sample Input 1	Sample Output 1
2016 9 28 3 1	happy

Sample Input 2	Sample Output 2
2016 10 21 5 0	unhappy