

## Mục lục

|                   |    |
|-------------------|----|
| TAXI . . . . .    | 1  |
| DOG . . . . .     | 3  |
| BRORDER . . . . . | 4  |
| STRkARY . . . . . | 5  |
| BGRAPH . . . . .  | 6  |
| BALOON . . . . .  | 7  |
| WORDS . . . . .   | 8  |
| PLUS . . . . .    | 9  |
| PMONEY . . . . .  | 10 |
| TRIPLE . . . . .  | 11 |
| DRONE . . . . .   | 12 |
| BEAUSTR . . . . . | 13 |

Nộp bài tại: 103.74.123.38:8888

Username: [Tên Đội] [Họ Tên viết gọn]. Ví dụ: Nguyễn Văn An đội Dak Lak -> DakLak NVAn

Password như username.

## Bài 1. TAXI

Một taxi tải phải phục vụ  $N$  yêu cầu  $1, \dots, N$  trong đó yêu cầu  $i$  có điểm đón  $i$  và điểm trả hàng  $i + N$ . Taxi xuất phát từ điểm 0, phục vụ  $N$  yêu cầu trên và quay trở lại điểm 0. Khoảng cách giữa các điểm được cho bởi một ma trận  $d_{(2N+1) \times (2N+1)}$  trong đó  $d_{i,j}$  là khoảng cách từ điểm  $i$  tới điểm  $j$  ( $0 \leq i, j \leq 2N$ ). Hãy tìm hành trình của taxi không được lặp lại đỉnh trừ đỉnh xuất phát và có tổng khoảng cách nhỏ nhất. Biết rằng giá trị này không vượt quá  $10^9$ .

## Dữ liệu vào

- Dòng 1:  $2N$  ( $1 \leq N \leq 10$ )
- Dòng  $i + 1$  ( $i = 1, \dots, 2N + 1$ ) là dòng  $i^{th}$  của ma trận khoảng cách.

## Kết quả

Ghi ra tổng khoảng cách của hành trình tìm được.

## Ví dụ

| test   | answer |
|--|--------|
| 6<br>0 4 2 3 5 4 4<br>4 0 7 5 2 3 1<br>3 2 0 1 2 1 9<br>2 3 5 0 9 8 3<br>2 1 4 6 0 9 1<br>9 8 1 4 2 0 8<br>1 2 3 2 5 4 0 | 13     |

## Explanations

Hành trình có khoảng cách nhỏ nhất là: 0 2 3 1 5 4 6 với tổng khoảng cách  $= 2 + 1 + 3 + 3 + 2 + 1 + 1 = 13$

## Bài 2. DOG

Khu vực bạn sống có rất nhiều chó. Chúng rất thích sửa, đặc biệt chúng còn đua nhau sửa khi nghe thấy một con nào đó sửa. Các chú chó trong vùng được đánh số từ 1 đến  $d$  ( $1 \leq d \leq 1000$ ). Mỗi con chó có thể nghe thấy tiếng sửa của một số con khác. Một chú chó  $j$  nghe thấy tiếng sửa của chú chó  $i$  nhưng chưa chắc chú chó  $i$  đã nghe thấy tiếng sửa của chú chó  $j$ . Mỗi khi nghe thấy tiếng sửa đầu tiên, chú chó  $j$  sẽ chờ đợi một thời gian  $w_j$  rồi mới cất tiếng sửa theo và sửa đúng một giây. Ban đầu tất cả các chú chó đều ngủ trừ chú chó số 1. Tại thời điểm giây thứ 0 chú chó 1 cất tiếng sửa (và sửa đúng một giây). Giả sử chú chó  $i$  bắt đầu sửa tại giây thứ  $n$  và chú chó  $j$  nghe thấy tiếng sửa của chú chó  $i$ , khi đó chú chó  $j$  sẽ tỉnh dậy ở giây thứ  $n$ , chờ đợi và sẽ bắt đầu sửa ở giây thứ  $n + w_j$ , sửa đúng 1 giây và đến giây thứ  $n + w_j + 1$  thì chú chó  $j$  nằm ngủ. Trong lúc chờ đợi chuẩn bị sửa hay lúc đang sửa, mọi tiếng sửa khác mà nó nghe thấy được đều bị bỏ qua.

**Yêu cầu:** Hãy xác định trong khoảng thời gian từ đầu đến hết giây thứ  $t$  ( $1 \leq t \leq 1000$ ) mỗi chú chó sửa bao nhiêu lần.

### Dữ liệu vào

- Dòng đầu tiên chứa số nguyên  $d$ ,
- Dòng thứ  $i$  trong  $d$  dòng sau chứa số nguyên  $w_i$  ( $1 \leq w_i \leq 1000$ ),
- Dòng tiếp theo chứa số nguyên  $f$  ( $1 \leq f \leq 10000$ ),
- Mỗi dòng trong  $f$  dòng tiếp theo chứa 2 số nguyên  $i$  và  $j$ , cho biết chú chó  $j$  có thể nghe được tiếng sửa của chú chó  $i$ ,
- Dòng cuối cùng chứa số nguyên  $t$ .

### Kết quả

Đưa ra  $d$  số nguyên trên  $d$  dòng, số ở dòng thứ  $i$  cho biết số lần chú chó  $i$  sửa.

### Ví dụ

| test | answer |  |
|------|--------|--|
| 3    | 3      |  |
| 1    | 2      |  |
| 1    | 2      |  |
| 3    |        |  |
| 3    |        |  |
| 1 2  |        |  |
| 2 3  |        |  |
| 3 1  |        |  |
| 10   |        |  |

| Giây thứ | Mô tả                            |
|----------|----------------------------------|
| 0        | chó 1 sửa, chó 2 tỉnh            |
| 1        | chó 1 ngủ, chó 2 sửa, chó 3 tỉnh |
| 2        | chó 2 ngủ                        |
| 3        | -                                |
| 4        | chó 3 sửa, chó 1 tỉnh            |
| 5        | chó 3 ngủ, chó 1 sửa, chó 2 tỉnh |
| 6        | chó 1 ngủ, chó 2 sửa, chó 3 tỉnh |
| 7        | chó 2 ngủ                        |
| 8        | -                                |
| 9        | chó 3 sửa, chó 1 tỉnh            |
| 10       | chó 3 ngủ, chó 1 sửa, chó 2 tỉnh |

### Giải thích ví dụ

## Bài 3. BRORDER

Các dấu ngoặc xuất hiện rất nhiều trong các biểu thức toán học để thể hiện thứ tự tính toán. Giờ đây ta bỏ hết các hạng tử toán tử đi, chỉ giữ lại các dấu ngoặc, biểu thức mà ta thu được gọi là một dãy ngoặc đúng. Cụ thể hơn:

- Xâu rỗng là biểu thức ngoặc đúng
- Nếu  $A$  là biểu thức ngoặc đúng thì  $(A)$  cũng là dãy ngoặc đúng
- Nếu  $A$  và  $B$  là biểu các thức ngoặc đúng thì  $AB$  cũng là biểu thức ngoặc đúng

Cho  $S$  là một biểu thức ngoặc đúng. Tiến hành sắp xếp tất cả các dãy ngoặc đúng độ dài  $|S|$  theo thứ tự từ điển (với ' $($ ' < ')'), sau đó đánh số thứ tự cho các xâu bắt đầu từ 1. Tìm số được đánh cho  $S$ . Bạn cần trả lời nhiều truy vấn như vậy, độ dài xâu  $S$  ở các truy vấn có thể khác nhau

### Dữ liệu vào

- Dòng đầu tiên chứa  $Q$  là số lượng truy vấn
- $Q$  dòng tiếp theo mỗi dòng chứa một xâu  $S$

### Kết quả

Ghi ra  $Q$  dòng, mỗi dòng chứa số nguyên dương là thứ tự của  $S$  với test tương ứng, chỉ cần in ra phần dư khi chia cho  $10^9 + 7$

### Ví dụ

| test   | answer |
|--------|--------|
| 3      | 5      |
| ()()() | 1      |
| ()     | 2      |
| ()()   |        |

### Giải thích

Ở truy vấn đầu tiên, các dãy ngoặc được đánh số là: 1: ((( ))) 2: (()()) 3: ()()() 4: ()(()) 5: ()()()

### Hạn chế

- $1 \leq |S|, Q \leq 10^6$ , tổng độ dài tất cả các xâu  $S$  không quá  $10^6$
- Subtask 1:  $Q \leq 10, |S| \leq 18$
- Subtask 2:  $|S| \leq 18$
- Subtask 3:  $|S| \leq 1000$ , tất cả các xâu  $S$  có cùng độ dài
- Subtask 4:  $|S| \leq 1000$
- Subtask 5: Ràng buộc gốc

## Bài 4. STRkARY

Xâu  $x$  được gọi là *chuyển dịch* của chuỗi  $y$ , nếu tồn tại hai chuỗi  $u, v$  ( $u, v$  có thể là chuỗi rỗng) sao cho  $x = uv$  và  $y = vu$ . Ở đây  $uv$  ký hiệu phép toán ghép  $v$  nối tiếp vào sau chuỗi  $u$ .

Ví dụ: Xâu  $x = \text{«abba»}$  là chuyển dịch của chuỗi  $y = \text{«baab»}$ , bởi vì với  $u = \text{«ab»}, v = \text{«ba»}$  ta có  $x = uv$  và  $y = vu$ .

Cho chuỗi ký tự  $s$  độ dài  $n$ . Chuỗi  $s$  được gọi là có tính chất  *$k$ -phân*, nếu ta có thể chia chuỗi  $s$  thành  $k$  chuỗi con liên tiếp có độ dài bằng nhau, sao cho mỗi chuỗi con trong chúng đều là chuyển dịch của một chuỗi con khác trong số  $k - 1$  chuỗi con còn lại. Ta quy ước rằng tất cả các chuỗi  $s$  đều có tính chất 1-phân.

Yêu cầu: Hãy liệt kê tất cả giá trị  $k$  sao cho chuỗi  $s$  đã cho có tính chất  $k$ -phân.

### Dữ liệu vào

Một dòng duy nhất chứa chuỗi  $s$  ( $1 \leq |s| \leq 200\,000$ ) gồm các ký tự latin thường.

### Kết quả

Dòng thứ nhất ghi ra số  $m$  là số lượng các giá trị  $k$  sao cho chuỗi  $s$  có tính chất  $k$ -phân.

Dòng thứ hai ghi ra  $m$  giá trị  $k_i$  nêu trên theo thứ tự tăng dần. Hai giá trị liên tiếp được ghi cách nhau bởi dấu cách.

### Ví dụ

| test     | answer     |
|----------|------------|
| abbabaab | 3<br>1 2 4 |
| abbababa | 2<br>1 4   |

### Lưu ý

Trong ví dụ thứ nhất chúng ta có thể giữ nguyên chuỗi ( $k = 1$ ); hoặc chia thành  $k = 2$  chuỗi con «abba» và «baab»; hoặc chia thành  $k = 4$  chuỗi con «ab», «ba», «ba» và «ab»..

## Bài 5. BGRAPH

Cho đơn đồ thị vô hướng  $G$  gồm  $n$  đỉnh và  $m$  cạnh (đồ thị không có khuyên và cạnh lặp). Bob muốn gán cho mỗi đỉnh một giá trị 1, 2 hoặc 3 sao cho tổng 2 số bất kì ở 2 đầu mút của 1 cạnh là số lẻ. Hiển nhiên đây là bài toán dễ và Bob tìm được lời giải ngay sau khi code thuật toán Brute Force duyệt toàn bộ các khả năng trên siêu máy tính của mình. Bob thắc mắc liệu có bao nhiêu cách đánh số thỏa mãn. Hai cách đánh số được coi là khác nhau nếu tồn tại một đỉnh được đánh số khác nhau ở trong hai cách. Anh ấy đưa ra câu đố này cho các bạn thi môn Phân tích và thiết kế thuật toán. Vì kết quả có thể rất lớn nên anh ấy chỉ cần in ra kết quả theo modul 998244353.

### Dữ liệu vào

- Dòng đầu bao gồm 2 số  $n$  và  $m$  là số đỉnh và số cạnh của đồ thị đã cho.
- $m$  dòng sau mỗi dòng chứa 2 số  $u$  và  $v$  cho biết có cạnh nối giữa  $u$  và  $v$ .

### Kết quả

In ra kết quả của bài toán.

### Ví dụ

| test       | answer |
|------------|--------|
| 2 1<br>1 2 | 4      |

### Hạn chế

- $1 \leq n \leq 3 \times 10^5$ ;  $0 \leq m \leq 3 \times 10^5$ .
- Có 50% test với  $n \leq 10$ .
- Có 20% test đồ thị liên thông.

## Bài 6. BALOON

Có  $n$  quả bóng bay, mỗi quả có một trong số 4 màu  $\{R, B, G, Y\}$  xếp thành một cột. Ở mỗi bước được phép chọn thủng dây bóng cùng màu liên tiếp nhau từ 2 quả trở lên. Nếu dây bị chọn thủng có  $k$  quả thì điểm số nhận được ở bước đi đó là  $k^2$ . Các quả bóng ở trên sẽ rơi xuống lấp chỗ trống trong cột sao cho thứ tự trên dưới ban đầu của các quả bóng không bị thay đổi. Ví dụ, cột ban đầu có 10 bóng, tính từ trên xuống cột có cấu hình  $BBGGBBYYYY$ . Để chọn thủng hết dây bóng, ta có thể chọn các quả liên tiếp  $BB, GG, BB, YYY$ . Khi đó, điểm số nhận được sẽ là  $2^2 + 2^2 + 2^2 + 3^2 = 21$ . Tuy nhiên, ta có cách khác để nhận được tổng điểm số cao hơn là  $GG, BBBB, YYY$ , với tổng điểm là  $2^2 + 4^2 + 3^2 = 29$ .

**Yêu cầu:** Cho xâu  $S$  độ dài  $n$  chỉ chứa các ký tự trong tập  $\{R, B, G, Y\}$  biểu diễn màu các quả bóng tính từ trên xuống dưới. Hãy xác định tổng điểm số lớn nhất nhận được khi phá hết bóng trong cột. Nếu không thể phá hết bóng thì tổng điểm là 0.

### Dữ liệu vào

Dòng đầu tiên ghi một số nguyên dương  $T \leq 5$  là số lượng test.

Mỗi dòng trong số  $T$  dòng tiếp theo chứa một xâu  $S$ .

### Kết quả

Mỗi dòng ghi ra một số nguyên là kết quả tìm được tương ứng với test trong dữ liệu vào.

### Ví dụ

| test       | answer |
|------------|--------|
| 4          | 29     |
| BBGGBBYYYY | 13     |
| BGGGB      | 24     |
| BGGBGGGG   | 0      |
| GBGB       |        |

### Hạn chế

- **Subtask 1 (28 điểm)**  $n \leq 15$ ;
- **Subtask 2 (16 điểm)**  $n \leq 200$ , chỉ có 2 màu  $G, B$  và độ dài các đoạn màu giống nhau liên tiếp là như nhau;
- **Subtask 3 (56 điểm)**  $n \leq 200$ .

## Bài 7. WORDS

File dữ liệu vào: `stdin`  
 File kết quả: `stdout`  
 Hạn chế thời gian: 0.5 giây  
 Hạn chế bộ nhớ: 512 M

Trong ngôn ngữ nhị phân  $\Omega$ , chỉ có 2 chữ cái là 0 và 1. Mỗi từ chính là 1 xâu nhị phân. Ký hiệu  $D$  là một tập các từ (1 tập các xâu nhị phân cho trước) của ngôn ngữ  $\Omega$ . Cho bảng vuông  $N \times N$ , mỗi ô của bảng điền giá trị 0 hoặc 1. Đếm trên các dòng của bảng vuông có bao nhiêu từ của ngôn ngữ  $\Omega$ .

Ví dụ: tập  $D = \{001, 101, 011\}$ , trong bảng sau sẽ có 4 từ của ngôn ngữ  $\Omega$  được đánh dấu.

|   |   |   |   |
|---|---|---|---|
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 |

### Dữ liệu vào

Bao gồm các dòng sau:

- Dòng 1: ghi số nguyên dương  $N$  ( $1 \leq N \leq 100$ );
- Dòng  $i + 1$  ( $i = 1, \dots, N$ ): ghi dòng thứ  $i$  của bảng (gồm  $N$  giá trị 0 hoặc 1 cách nhau bởi dấu cách);
- Dòng  $N + 2$ : ghi số nguyên dương  $k$  ( $1 \leq k \leq 10^5$ ) là tổng số các từ trong tập  $D$ ;
- Dòng  $N + 2 + i$  ( $i = 1, \dots, k$ ): ghi 1 xâu chỉ gồm các ký tự 0 hoặc 1 viết liền nhau (độ dài mỗi xâu nhỏ hơn hoặc bằng 50).

### Kết quả

Dòng duy nhất ghi số từ của ngôn ngữ  $\Omega$  xuất hiện trên các dòng của bảng vuông đã cho.

### Ví dụ

| stdin   | stdout |
|---------|--------|
| 4       | 4      |
| 1 0 0 1 |        |
| 1 1 1 1 |        |
| 1 0 1 1 |        |
| 0 0 0 1 |        |
| 3       |        |
| 001     |        |
| 011     |        |
| 101     |        |



## Bài 8. PLUS

File dữ liệu vào: `stdin`  
File kết quả: `stdout`  
Hạn chế thời gian: 1 giây

Cho hai số nguyên  $A$  và  $B$ . Cần biến đổi để hai số này bằng nhau bằng cách thực hiện không quá 5000 lệnh, mỗi lệnh có dạng " $A += B$ " hoặc " $B += A$ " với ý nghĩa là "tăng  $A$  lên một lượng bằng  $B$ " hoặc "Tăng  $B$  lên một lượng bằng  $A$ " (hoặc " $A += A$ " hoặc " $B += B$ ").

### Dữ liệu vào

- Gồm hai số nguyên  $A$   $B$

### Kết quả

- Dòng đầu ghi số lượng biến đổi  $Q$
- Theo sau là  $Q$  dòng, mỗi dòng ghi một biến đổi

### Ví dụ

| stdin | stdout                            |
|-------|-----------------------------------|
| 2 3   | 4<br>B+=B<br>B+=A<br>A+=A<br>A+=A |

### Hạn chế

- $1 \leq A, B \leq 10^{18}$

## Bài 9. PMONEY

File dữ liệu vào: `stdin`  
File kết quả: `stdout`  
Hạn chế thời gian: 1 giây

Sau đây tôi sẽ chỉ cho các bạn cách để mua được ô tô nhờ tiết kiệm tiền ăn sáng: Mỗi ngày bạn để dành ra 10k thì một tháng sẽ có 300k. Như vậy sau mười tháng bạn đã có trong tay 3M, xin bố mẹ 100M nữa là đủ tiền mua ô tô.

Nghe có vẻ khả thi nhưng Hùng đã hối hận vì nghĩ đến việc sẽ bỏ bữa sáng nhiều như vậy. Thay vào đó, cậu quyết định sẽ tiết kiệm tiền tiêu vặt hàng ngày. Mỗi ngày, mẹ của Hùng sẽ cho cậu một số tiền đều đặn vào sáng sớm. Lượng tiền mà hàng ngày mẹ đưa cho cậu (ký hiệu là  $x$ ) có thể thay đổi. Ban đầu  $x = 0$ , đến buổi tối của mỗi ngày,  $x$  sẽ tăng lên một, giảm đi một hoặc giữ nguyên, tùy thuộc vào mức độ ngoan ngoãn của cậu trong ngày đó. Vì tất cả tiền đều nhờ mẹ giữ hộ nên Hùng không biết chính xác là cậu đang có bao nhiêu tiền, cậu cố nhớ lại sự thay đổi của  $x$  trong suốt thời gian qua, tuy nhiên có một số ngày cậu không thể nhớ chính xác. Một điều mà cậu nhớ chính xác đó là sau ngày cuối cùng thì  $x = 0$  và tại mọi thời điểm thì  $x \geq 0$ . Hãy giúp Hùng xác định số tiền ít nhất và nhiều nhất cậu có thể có, hoặc thông báo là cậu đã nhớ nhầm (nếu không có cách khôi phục nào thỏa mãn các ràng buộc của  $x$ ). Dù sao đi nữa, bữa sáng cũng rất quan trọng và bạn nên có kế hoạch để sớm mua được ô tô :D

### Dữ liệu vào

- Gồm một xâu chứa không quá  $10^6$  ký tự thuộc tập  $\{+, -, 0, _\}$  tương ứng là  $x$  tăng, giảm, giữ nguyên hoặc không nhớ rõ

### Kết quả

- In ra "NIE" Nếu Hùng nhớ nhầm, ngược lại ghi hai số nguyên là số tiền ít nhất và nhiều nhất mà cậu đang có

### Ví dụ

| stdin                       | stdout            |
|-----------------------------|-------------------|
| <code>+_ + -0_0_ + -</code> | <code>3 13</code> |
| <code>-- -- -- -</code>     | <code>NIE</code>  |

### Hạn chế

- Có 50% số test với độ dài xâu không quá 1000

## Bài 10. TRIPLE

Cho một dãy  $s$  gồm  $N$  số nguyên dương  $a_1, \dots, a_N$  và một số nguyên dương  $M$ . Hãy tính số lượng  $Q$  bộ ba  $(i, j, k)$  sao cho  $1 \leq i < j < k \leq N$  và  $a_i + a_j + a_k = M$ .

### Dữ liệu vào

- Dòng 1 chứa  $N$  và  $M$  ( $1 \leq N \leq 10^4$ ,  $1 \leq M \leq 10^9$ ).
- Dòng 2 chứa  $N$  số nguyên dương  $a_1, \dots, a_N$  của dãy  $s$ .

### Kết quả

Ghi ra phần dư của phép chia  $Q$  cho  $10^9 + 7$ .

### Ví dụ

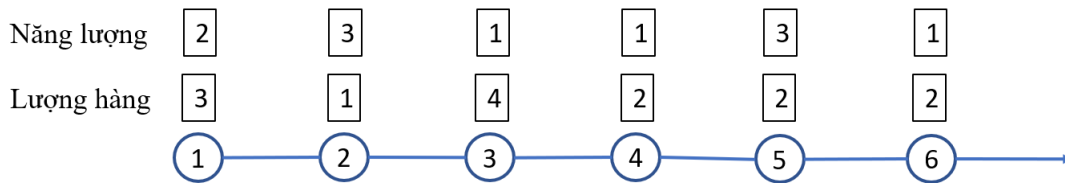
| test              | answer | Explanation   |
|-------------------|--------|---|
| 5 10<br>3 2 4 5 1 | 2      | Có 2 bộ ba có tổng bằng 10 là (3, 2, 5) và (4, 5, 1). |

### Hạn chế

- 50% số điểm thỏa mãn  $M \leq 10^4$
- 25% số điểm thỏa mãn  $M \leq 10^6$
- 25% số điểm thỏa mãn  $M \leq 10^9$

## Bài 11. DRONE

Một thiết bị bay Drone có nhiệm vụ bay qua các điểm  $1, 2, \dots, N$  được phân bố đều trên 1 đường thẳng, điểm  $i$  có tọa độ  $i$ . Mỗi điểm  $i$  vừa có hàng hóa cần lấy với lượng hàng là  $c_i$  và vừa là trạm để nạp năng lượng với lượng năng lượng  $a_i$ . Khi Drone dừng tại trạm  $i$  thì nó sẽ lấy được lượng hàng là  $c_i$  và nạp đúng mức năng lượng  $a_i$  bất kể là đang còn thừa năng lượng hay không. Tiếp theo nó có thể bay tiếp tối đa đến trạm  $i + a_i$  (nó có thể dừng tiếp theo ở trạm nào đó trong số các trạm  $i + 1, i + 2, \dots, i + a_i$ ). Do đặc tính kỹ thuật, Drone chỉ có thể dừng tối đa  $K$  trạm. Hãy tính toán cách đi cho Drone xuất phát từ điểm 1 sao cho nó đến được điểm cuối  $N$  và lấy được nhiều hàng hóa nhất. Ví dụ: với thông tin của 6 điểm như sau:



thì Drone sẽ xuất phát từ điểm 1 và dừng tại điểm 2, điểm 5, và điểm 6, tổng lượng hàng lấy được là  $3+1+2+2 = 8$ .

### Dữ liệu vào

Dữ liệu đầu vào bao gồm các dòng sau:

- Dòng 1: chứa  $N$  và  $K$  ( $1 \leq N \leq 3000, 1 \leq K \leq 100$ )
- Dòng 2:  $N$  số nguyên dương  $c_1, c_2, \dots, c_N$  (Giá trị các số từ 1 đến 20)
- Dòng 3:  $N$  số nguyên dương  $a_1, a_2, \dots, a_N$  (Giá trị các số từ 1 đến 50)

### Kết quả

Ghi ra tổng lượng hàng lớn nhất mà Drone lấy được hoặc ghi giá trị -1 nếu không có cách đi thỏa mãn yêu cầu đặt ra.

### Ví dụ

| test                              | answer |
|-----------------------------------|--------|
| 6 3<br>3 1 4 2 2 2<br>2 3 1 1 3 1 | 8      |

## Bài 12. BEAUSTR

Xâu nhị phân là chuỗi chỉ chứa các ký tự 0 hoặc 1. Một chuỗi nhị phân được gọi là *đẹp* nếu với mỗi ký tự 1 trong chuỗi, số lượng ký tự 0 liên tiếp từ nó tới ký tự 1 gần nhất bên trái hoặc tới đầu chuỗi bằng số lượng ký tự 0 liên tiếp từ nó tới ký tự 1 gần nhất bên phải hoặc tới cuối chuỗi. Tức là với mỗi số 1, số lượng ký tự 0 liên tiếp ngay bên trái nó bằng số lượng ký tự 0 liên tiếp ngay bên phải nó. Ví dụ chuỗi 0001000 là chuỗi đẹp, còn chuỗi 001010 là không đẹp vì bên trái ký tự 1 đầu tiên có hai ký tự 0 và bên phải nó chỉ có một ký tự 0. Cho trước một chuỗi nhị phân bất kỳ, bạn được phép xóa một số ký tự trong đó để biến nó thành chuỗi đẹp.

**Yêu cầu:** Hãy in ra độ dài của chuỗi đẹp dài nhất có thể sinh ra bằng cách xóa các ký tự của một chuỗi cho trước.

### Dữ liệu vào

Dòng đầu tiên chứa số nguyên  $n$  là độ dài của chuỗi nhị phân ( $1 \leq n \leq 500\,000$ ). Dòng thứ 2 là một dãy có  $n$  ký tự 0 hoặc 1. Đảm bảo dòng này chứa ít nhất một ký tự 1.

### Kết quả

Ghi ra một dòng chứa một số nguyên duy nhất là độ dài của chuỗi đẹp dài nhất có thể thu được.

### Hạn chế

Ký hiệu  $k$  là số lượng ký tự 1 trong chuỗi.

- Subtask 1 (25 điểm):  $n \leq 1000, k \leq 2$ ;
- Subtask 2 (25 điểm):  $n \leq 1000, k \leq 15$ ;
- Subtask 3 (20 điểm):  $n \leq 1000$ ;
- Subtask 4 (12.5 điểm):  $n \leq 100\,000, k \leq 50$ ;
- Subtask 5 (17.5 điểm):  $n \leq 500\,000$ .

### Ví dụ

| test             | answer |
|------------------|--------|
| 10<br>0000010100 | 7      |
| 4<br>1111        | 4      |
| 7<br>0101001     | 5      |

### Lưu ý

Từ chuỗi 0000010100 ta có thể thu được chuỗi đẹp dài nhất là 0001000 bằng cách xóa ký tự 1 phía sau và 2 ký tự 0 đầu tiên. Chuỗi 1111 bản thân là chuỗi đẹp nên không cần xóa ký tự nào. Từ chuỗi 0101001 ta có thể xóa hai ký tự cuối cùng để thu được chuỗi đẹp 01010.