パソコン甲子園2026

全国高等学校パソコンコンクール
プログラミング部門 予選問題

２０２６年9月１３日（日）　１３：３０～１６：３０



全国高等学校パソコンコンクール実行委員会

問題１　２０２６　（１点）

この問題は、プログラムの提出を練習するための問題である。通常の問題では、入力を受け取り何らかの処理を行った結果を出力するが、この問題では入力は与えられない。

課題

2026 を出力するプログラムを作成せよ。

入力

入力は与えられない。

出力

2026 を１行に出力する。

入出力例

入力例	出力例
	2026

問題２　ポイント強化月間　（２点）

イヅア市では商品購入のときに akaBeko カードを見せると、商品に応じたポイントをもらうことができる。今月はポイント強化月間で、通常の月の３倍のポイントがもらえる。さらに、今月は akaBeko カード創設１０周年にあたるため、１回の利用ごとに１０ポイントが加算される。

課題

商品を通常の月に１回購入したときにもらえるポイントが与えられる。同じ商品を今月１回購入したとき、通常の月と比べて追加でもらえるポイントを出力するプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

p

１行に、商品を通常の月に１回購入したときにもらえるポイント$p(1 \leq p \leq 10{,}000)$が与えられる。

出力

追加でもらえるポイントを１行に出力する。

入出力例

入力例１	出力例１
100	210

通常の月には100ポイントもらえ、今月は310ポイントもらえるので、追加でもらえるポイントは210である。

入力例２	出力例２
5000	10010

問題３　チェス盤距離　（３点）

座標平面上の２点間の距離は、それらの点の座標がそれぞれ(x_1, y_1)と(x_2, y_2)のとき、通常は$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$で与えられる。しかし、他にもいろいろな距離がある。たとえば、チェス盤距離という名前で知られる距離は、$x_1 - x_2$の絶対値と$y_1 - y_2$の絶対値の大きい方の値で与えられる。

課題

座標平面上の２点の座標が与えられたとき、２点間のチェス盤距離を出力するプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

$x_1\ y_1\ x_2\ y_2$

１行に、１つ目の点の座標を表す整数$x_1, y_1(-1{,}000 \leq x_1, y_1 \leq 1{,}000)$と２つ目の点の座標を表す整数$x_2, y_2(-1{,}000 \leq x_2, y_2 \leq 1{,}000)$が与えられる。

出力

２点間のチェス盤距離を１行に出力する。

入出力例

入力例１	出力例１
0 0 -500 300	500

入力例２	出力例２
30 10 -30 80	70

入力例３	出力例３
100 50 80 130	80

問題４　積みゲー　（４点）

PCK 君はパッケージ版のゲームを買うと、そのゲームソフトのケースを机の上に積んでいく。新しいゲームを買ったときは、すでに積んであるケースがあればその一番上にケースを置き、１つもなければ机の上に置く。

このように積まれたゲームの山を「積みゲー」と呼ぶことがある。ゲームを遊ぶときは、積みゲーの一番上に積んであるゲームを１つ取り、そのゲームを遊ぶ。遊び終えたゲームは別の棚に片付けるため、積みゲーの山からはなくなる。また、PCK 君はときどき、積みゲーの山にゲームがいくつ積まれているかを確認する。

課題

PCK 君の積みゲーに関する操作が与えられたとき、操作を順に処理し、必要に応じてゲームの番号またはゲームの数を出力するプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

```
Q
op_1
op_2
⋮
op_Q
```

１行目に、積みゲーに関する行動の回数$Q(1 \leq Q \leq 100{,}000 = 10^5)$が与えられる。続く$Q$行に、$i$番目の操作$op_i$が与えられる。各$op_i$は、以下のいずれかの形式で与えられる。

```
1 x
```

または

```
2
```

または

```
3
```

$1\ x$は、ゲーム番号が$x(1 \leq x \leq 100{,}000 = 10^5)$であるゲームを、積みゲーの一番上に積む操作を表す。2は、積みゲーの一番上に積まれているゲームを１つ取り出してプレイし、そのゲームの番号を答える操作を表す。ただし、積みゲーにゲームが１つも積まれていないときに、この操作が与えられることはない。3は、現在、積みゲーに積まれているゲームの数を答える操作を表す。操作2または操作3は、少なくとも１回は与えられる。

出力

操作2または操作3が与えられるたびに１行に結果を出力する。操作2が与えられたときはゲームの番号を、操作3が与えられたときはゲームの数を、それぞれ出力する。

入出力例

入力例	出力例
9 1 123 1 456 3 2 3 1 789 2 2 3	2 456 1 789 123 0

問題５　満員のシャトルバス　（６点）

パソコン甲子園の本選会場へ向かうため、N人の選手を１台のシャトルバスに乗車させたい。以下の手順に従って、選手をバスに乗車させる。ただし、最初は誰も乗っていない座席数１のバスが停まっている。

1. 選手がバスの座席に座る。座席１つには選手１人が座れる。選手が全員着席できたら、乗車は完了する。選手が全員着席できなければ、次の手順に移る。
2. 現在の座席数の２倍の座席数を持つバスを手配する。手配したバスには誰も乗っていない。
3. 手配したバスが停留所に到着したら、それまでのバスに乗っていたすべての選手を、手配したバスへ移動させ着席させる。このとき、選手１人が移動するごとに１回の乗り換えが発生する。
4. それまでのバスは役目を終えて去る。まだ着席できていない選手がいるので、1 から繰り返す。

乗車が完了するまでに発生する選手の乗り換えの回数の総和を求めたい。

課題

選手の人数Nが与えられたとき、乗車が完了するまでに発生する選手の乗り換えの回数の総和を求めるプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

```
N
```

１行に、選手の人数N($1 \leq N \leq 1{,}000$)が与えられる。

出力

乗車が完了するまでに発生する選手の乗り換えの回数の総和を１行に出力する。

入出力例

入力例１	出力例１
4	3

４人の選手が順に乗車する際の、座席数と乗り換え回数の遷移は以下の通り。

- １人目: 最初に停まっている座席数１のバスに乗る。乗り換えは発生せず、座席数は１で空席数は０になる。
- ２人目: 空席が無いため、座席数１に対して座席数が２倍である座席数２のバスを手配。それまでのバスにいた１人が移動するため、乗り換え１回。座席数は２で空席数は０になる。
- ３人目: 空席が無いため、座席数２に対して座席数が２倍である座席数４のバスを手配。それまでのバスにいた２人が移動するため、乗り換え２回。座席数は４で空席数は１になる。
- ４人目: 空席が１つあるため、バスの手配は起きない。乗り換えは０回。座席数は４で空席数は０になる。

よって、乗り換え回数の総和は 0 + 1 + 2 + 0 = 3 となる。

入力例２	出力例２
9	15

4 人目までは入出力例１と同じ（４人目までで乗り換え計３回、座席数４、空席数０）なので省略する。５人目以降の遷移は以下の通り。

- ５人目: 空席が無いため、座席数４に対して座席数が２倍である座席数８のバスを手配。それまでのバスにいた４人が移動するため、乗り換え４回。座席数は８で空席数は３になる。
- ６～８人目: 空席があるため、バスの手配は起きない。乗り換えは０回。８人目が乗車した時点で、座席数は８で空席数は０になる。
- ９人目: 空席が無いため、座席数８に対して座席数が２倍である座席数１６のバスを手配。それまでのバスにいた８人が移動するため、乗り換え８回。座席数は１６で空席数は７になる。

よって、乗り換え回数の総和は 3 + 4 + 0 + 8 = 15 となる。

問題の背景：ならし計算量（Amortized Complexity）

本問題は、多くのプログラミング言語で採用されている動的配列のメモリ管理手法そのものである。配列の容量が足りなくなるたびに要素を全コピーする操作は、単一の操作で見れば最悪 $O(N)$ の時間がかかる重い処理である。しかし、容量を「倍々」に増やしていくことで、拡張が発生する頻度を劇的に減らすことができる。

実際に解答プログラムを実行することで、要素をN回追加したときのコピー回数の総和は、常にNの約２倍（最大でも$2N$未満）の定数倍に収まっていることが確認できる。これは、「N回の要素の追加にかかる合計コストが$O(N)$である」ことを意味し、１回あたりの平均コストに直すと $O(N) / N = O(1)$になる。

このように、最悪ケースのコストは大きいものの、複数回の操作全体で平均したときの時間計算量を評価する手法をならし計算量と呼ぶ。動的配列は、メモリを少し多めに確保する負担を受け入れることで、実用上要素の追加にかかる時間を平均定数時間$O(1)$で高速に実行できる仕組みを実現している。

問題６　バスの駐車場所　（６点）

パソコン甲子園の本選会場に、選手を乗せたバスが到着する。会場の駐車場には、同じ大きさのN個の区画が左から右に向かって一列に並んでいる。それぞれの区画は、空いているか、すでに駐車済みかのいずれかである。

バスを駐車させるときは連続するL個の区画をすべて空ける必要がある。その連続するL個の区画の中にある駐車済みの区画がK個までなら、そこに駐車している車を移動させてスペースを空け、バスを駐車することができる。このとき、バスを駐車する方法がいくつあるかを知りたい。バスが使用する区画が１つでも異なる場合は、使用する区画の一部が重なっていても、異なる駐車方法として数える。

課題

駐車場の使用状況が与えられたとき、バスを駐車する方法の数を求めるプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

$N\ L\ K$ $s_1\ s_2\ \cdots\ s_N$

１行目に、駐車場の区画の数$N(1 \leq N \leq 100{,}000)$と、バスを駐車するために必要な区画の数$L(1 \leq L \leq N)$と、空けることができる区画の数$K(0 \leq K \leq L)$が与えられる。続く１行に、各区画の使用状況を表す整数s_iが与えられる。s_iは、左から i番目の区画が空いているとき0、使用されているとき1である。

出力

バスの駐車方法の数を１行に出力する。

入出力例

入力例１	出力例１
10 2 0 1 0 0 0 0 1 0 0 0 1	5

バスが使用する最も左の区画を、左から２番目、３番目、４番目、７番目、８番目のいずれかにすることができるので、答えは5である。

入力例２	出力例２
10 2 1 1 1 0 0 0 1 0 0 0 1	8

バスが使用する最も左の区画を、左から２番目、３番目、４番目、５番目、６番目、７番目、８番目、９番目のいずれかにすることができるので、答えは8である。左から１番目の区画をバスの最も左の区画とするときは、$K = 1$では左から１番目の区画と２番目の区画の２つとも空けることはできないので、バスを置くことはできない。

入力例３	出力例３
3 2 0 0 1 0	0

長さ２の区画を確保することができないので、答えは0である

問題７　配達の利益　（８点）

PCK 君は配達のアルバイトをしている。現在、N個の荷物が配達センターに集められている。これらの荷物の重さと、配達で得られる報酬は、荷物ごとに様々である。

１回の配達では、N個の荷物のうち、いくつかの荷物を選んでトラックに載せる。ただし、荷物を１つも選ばずに配達に行くこともできる。配達に使うトラックの荷台はバネにより支えられており、選んだ荷物の重さの合計がMまでであれば、バネは自然な状態で荷物を支えられる。しかし、選んだ荷物の重さの合計がMを超えると、(選んだ荷物の重さの合計 $-M$)^{2}のコストがかかる。選んだ荷物の重さの合計がMを超えない場合、コストは0である。得られる報酬の合計からこのコストを引いた値が配達の利益となる。

荷物の選び方で報酬とコストが変わるので、PCK 君は得られる利益が最大になるように荷物を選んだときの利益を求めたい。

課題

配達センターにある荷物の個数N、バネが自然な状態で荷物を支えられる重さの上限値M、それぞれの荷物の重さと配達の報酬が与えられたとき、得られる利益の最大値を出力するプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

$N\ M$
$w_1\ v_1$
$w_2\ v_2$
$\vdots$
$w_N\ v_N$

１行目に、配達センターにある荷物の数Nとバネが自然な状態で荷物を支えられる重さの上限値Mが与えられる$(1 \leq N, M \leq 1{,}000)$。続く$N$行に、荷物$i$の重さ$w_i(1 \leq w_i \leq 1{,}000{,}000{,}000 = 10^9)$と配達することで得られる報酬$v_i(1 \leq v_i \leq 100{,}000 = 10^5)$が与えられる。

出力

１行に、配達によって得られる利益の最大値を出力する。

入出力例

入力例	出力例
5 10 1 1 2 2 3 3 4 4 5 100	105

問題８　三角形の個数　（８点）

平面上にN個の点がある。これらの点から互いに異なる３点を選び、その３点を頂点とする三角形を考える。三角形を、３辺の長さに応じて次の３種類に分類する。

①　正三角形：３辺の長さがすべて等しい三角形
②　二等辺三角形：①ではなく、２辺の長さが等しい三角形
③　その他の三角形：①でも②でもない三角形

それぞれの種類の三角形がいくつあるかを求めたい。ただし、選んだ３点が一直線上にある場合は三角形とはみなさない。また、点を選ぶ順番だけが異なるものは、同じ三角形とみなす。

課題

N個の点の座標が与えられたとき、正三角形、二等辺三角形、その他の三角形の個数をそれぞれ求めるプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

N
$x_1\ y_1$
$x_2\ y_2$
$\vdots$
$x_N\ y_N$

１行目に、与えられる点の個数$N(3 \leq N \leq 300)$が与えられる。続くN行に、与えられる点のx, y座標$x_i, y_i(0 \leq x_i, y_i \leq 1{,}000{,}000{,}000 = 10^9)$が整数で与えられる。ただし、同じ座標の点は与えられない$(i \neq j$について、$x_i \neq x_j$または$y_i \neq y_j)$。

出力

正三角形、二等辺三角形、その他の三角形の個数を、この順にそれぞれ１行に出力する。

入出力例

入力例	出力例
4 0 0 0 2 0 3 1 1	0 1 2

問題９　魔法陣の調律　　（１２点）

魔女のマリーは、魔法陣へ効率よく魔力を供給する方法を研究している。魔法陣へ魔力を供給するには、一列に並んだ魔力石を操作する必要がある。それぞれの魔力石には、初めからある量の魔力が蓄えられている。

マリーは複数の魔導書を持っている。それぞれの魔導書には複数の魔法が記されている。魔導書を１回開くと、記されているすべての魔法が１回ずつ発動する。各魔法には、魔力を増やす範囲と増やす量が定められている。その魔法が発動すると、指定された範囲にあるすべての魔力石に蓄えられている魔力が、その量だけ増える。研究を進めるため、マリーは次の２種類の操作を何度も行う。

- 魔導書を１回開く。
- 指定した範囲の魔力石に蓄えられている魔力の合計を調べる。

しかし、操作の回数が多いため、手作業で処理することは難しい。そこで、マリーはこれらの操作を効率よく処理するプログラムを作成することにした。

課題

魔力石に蓄えられている魔力の初期値、各魔導書に記されている魔法、およびQ個の操作が与えられる。各操作を順に処理し、魔力の合計を調べる操作ごとに結果を出力するプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

$N\ M\ Q$
$A_1\ A_2\ \cdots\ A_N$
K_1
$l_{1,1}\ r_{1,1}\ v_{1,1}$
$l_{1,2}\ r_{1,2}\ v_{1,2}$
$\vdots$
$l_{1,K_1}\ r_{1,K_1}\ v_{1,K_1}$
K_2
$l_{2,1}\ r_{2,1}\ v_{2,1}$
$l_{2,2}\ r_{2,2}\ v_{2,2}$
$\vdots$
$l_{2,K_2}\ r_{2,K_2}\ v_{2,K_2}$
$\vdots$
K_M
$l_{M,1}\ r_{M,1}\ v_{M,1}$
$l_{M,2}\ r_{M,2}\ v_{M,2}$
$\vdots$
$l_{M,K_M}\ r_{M,K_M}\ v_{M,K_M}$
$operation_1$
$operation_2$
$\vdots$
$operation_Q$

１行目に、魔力石の個数$N(1 \leq N \leq 200{,}000 = 2 \times 10^5)$、魔導書の冊数$M(1 \leq M \leq 20)$、操作の回数$Q(1 \leq Q \leq 200{,}000 = 2 \times 10^5)$が与えられる。続く１行に、先頭から$i$番目の魔力石に蓄えられている魔

力の初期値A_i $(0 \leq A_i \leq 1{,}000)$が整数で与えられる。続いて、各魔導書に記されている魔法の情報が、１番目の魔導書から順に与えられる。i番目の魔導書について、最初の行に、その魔導書に記されている魔法の個数$K_i(1 \leq K_i \leq 200{,}000 = 2 \times 10^5)$が与えられる。続く$K_i$行に、$j$番目の魔法の情報が与えられる。魔法の情報は、３つの整数 $l_{i,j}, r_{i,j}, v_{i,j} (1 \leq l_{i,j} \leq r_{i,j} \leq N,\ 1 \leq v_{i,j} \leq 1{,}000)$で与えられる。これは、左から $l_{i,j}$番目から $r_{i,j}$番目までのすべての魔力石について、蓄えられている魔力をそれぞれ$v_{i,j}$増やす魔法を表す。ただし、K_iの総和は$200{,}000 = 2 \times 10^5$を超えない。

続くQ行に操作が与えられる。各操作は、次のいずれかの形式で与えられる。

```
1 x
```

この操作では、$x(1 \leq x \leq M)$番目の魔導書を１回開く。

```
2 p q
```

この操作では、pおよび$q(1 \leq p \leq q \leq N)$が与えられる。左から$p$番目から$q$番目までの魔力石に蓄えられている魔力の合計を調べる。魔力の合計を調べる操作は１回以上与えられる。

時間制限

入力に対して、実行時間が３秒を超えてはならない。

出力

魔力の合計を調べる操作１つに対して、調べた魔力の合計を、１行に１つずつ出力する。

入出力例

入力例	出力例
5 2 6 1 2 3 4 5 2 1 3 2 2 5 1 1 3 4 10 2 1 5 1 1 2 2 4 1 2 2 3 5 2 1 1	15 16 37 3

初めに各魔力石に蓄えられている魔力は、左から順に1，2，3，4，5である。そのため、最初の操作で調べる魔力の合計は15となる。次に、１番目の魔導書を１回開く。最初の魔法によって１番目から３番目までの魔力石の魔力がそれぞれ２増え、続いて次の魔法によって２番目から５番目までの魔力石の魔力がそれぞれ１増える。このとき、各魔力石に蓄えられている魔力は、左から順に3，5，6，5，6となる。そのため、２番目から４番目までの魔力の合計は16となる。その後、２番目の魔導書を１回開くと、３番目と４番目の魔力石の魔力がそれぞれ１０増える。各魔力石に蓄えられている魔力は、左から順に3，5，16，15，6となる。そのため、３番目から５番目までの魔力の合計は37、１番目の魔力石に蓄えられている魔力は3となる。

問題１０　狙われない道のり　（１４点）

パソベぇは、森の奥深くで見つけた迷宮に、うっかり足を踏み入れてしまった。迷宮は$H \times W$のマス目からなり、壁や弓兵が存在する。以下、上からi行目、左からj列目のマスをマス(i, j)と表す。各マスは次のいずれかである。

- 床：パソベぇが立ち入ることのできるマス。
- 壁：パソベぇが立ち入ることができず、矢も通過できないマス。
- 弓兵：上、下、左、右のいずれかの方向に弓を構えている。パソベぇが立ち入ることができず、矢も通過できないマス。

パソベぇは、現在いるマスから、上、下、左、右に隣接する床のマスを移動候補として選ぶことができる。ただし、後述する安全条件を満たす移動候補にのみ移動することができる。弓兵たちは移動したり、弓を向けている方向を変えたりすることはない。パソベぇは、壁のマスと、弓兵のいるマスを把握しているが、はじめは、それぞれの弓兵がどの方向に弓を向けているかを知らない。

床のマスＦと弓兵Ｂが同じ行または列にあり、マスＦと弓兵Ｂの間に壁も他の弓兵もないとき、マスＦから弓兵Ｂを見通せるものとする。パソベぇがある床のマスにいるとき、そのマスから見通せるすべての弓兵について、パソベぇはそれらの弓兵が弓を向けている方向を知る。一度判明した弓兵の方向は、その後も分かっている。マスＦから弓兵Ｂを見通すことができ、かつ、弓兵Ｂがマスの方向に弓を向けているとき、弓兵Ｂはマスを狙っているものとする。

パソベぇは頑丈な盾を持っており、1 人の弓兵から放たれる矢であれば何度でも防ぐことができる。しかし、2 人以上の弓兵から同時に矢を受けると、防ぎきることができない。そこでパソベぇは、移動する直前までに判明している情報をもとに、移動候補となるマスを狙っている可能性のある弓兵の人数を考える。移動候補から見通せる弓兵について、その弓兵が向いている方向がまだ判明していない場合、または、向いている方向が判明していてその移動候補を実際に狙っている場合、その弓兵はその移動候補を狙っている可能性があるものとする。

移動候補を狙っている可能性のある弓兵が 1 人以下であることを安全条件という。パソベぇは、安全条件を満たす移動候補にのみ移動することができる。移動候補に到着したことで新たに判明する弓兵の方向は、その移動候補が安全条件を満たすかどうかを判断する際には用いず、それより後の移動に用いる。

パソベぇは危険を避けつつ進みながら、迷宮の入口であるマス$(1, 1)$から出口であるマス(H, W)へたどり着きたい。果たして、そのような安全な道のりは存在するのだろうか。

課題

迷宮の行数Hと列数W、および迷宮の各マスの情報が与えられたとき、パソベぇが迷宮の入口から出口にたどり着くための安全な道のりが存在するかどうかを出力するプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

$H\ W$
$s_{1,1}s_{1,2}\cdots s_{1,W}$
$s_{2,1}s_{2,2}\cdots s_{2,W}$
$\vdots$
$s_{H,1}s_{H,2}\cdots s_{H,W}$

１行目に迷宮の行数Hと列数$W(2 \leq H, W \leq 100{,}000 = 10^5,\ H \times W \leq 200{,}000 = 2 \times 10^5)$が与えられる。続く$H$行に迷宮の情報が与えられる。$i$行目は床（.）、壁（#）、弓兵（^，v，<，>）のいずれかを表す記号からなる長さWの文字列$s_{i,1}s_{i,2}\cdots s_{i,W}$である。^，v，<，>は、それぞれ弓兵が上、下、左、右に弓を向けていることを表す。$s_{i,j}$はマス(i,j)の情報を表す。ただし、入口のマス$(1,1)$と出口のマス(H,W)は床である。また、入口のマス$(1,1)$から見通せる弓兵は１人以下である。

出力

安全な道のりが存在するなら「Yes」を、存在しないなら「No」を１行に出力する。

入出力例

入力例１	出力例１
4 5 .#.v. ....>.	Yes

(1, 1)→(2, 1)→(2, 2)→(2, 3)→(3, 3)→(4, 3)→(3, 3)→(3, 4)→(3, 5)→(4, 5)と移動すれば、狙っている可能性のある弓兵が2人以上いるマスへ移動せずに出口へたどり着ける。この道のりにおいて(4, 3)を訪れる前では、(1, 4)と(4, 4)の両方の弓兵から(3, 4)が狙われる可能性が残っているため、(3, 4)は安全条件を満たさず、移動することはできない。マス(4, 3)から弓兵(4, 4)を見通せるため、この弓兵が弓を向けている方向が判明する。その結果、(3, 4) を狙っている可能性があるのは(1, 4)の弓兵だけとなり、マス(3, 4) は安全条件を満たすため、移動できる。

入力例２	出力例２
3 5 .#v#. ..... .#^#.	No

入力例３	出力例３
3 5 .#^#. ..... .#v#.	No

マス(2, 3)を通れば出口にたどり着けるが、パソベぇは２人いる弓兵の向きが分からない。それぞれの弓兵が狙っている方向を安全に判明させる方法が無いためNoである。

入力例４	出力例４
3 3 .#. .#. .v.	No

問題１１　命令ログの検査　（１６点）

イヅア大学のヤエ教授は最新鋭の制御装置を開発した。この制御装置には、N個の命令があり、1 からNまでの番号が付けられている。このうち、1 からMまでの番号が付いた命令を標準命令と呼んでいる。

ジョー研究員はヤエ教授から制御装置の動作を検査するよう依頼された。検査モードで制御装置を実行すると、実行された命令の番号を順に記録した命令ログが出力される。ログには同じ命令番号が複数回現れることがある。

検査では、指定された２つの区間を命令ログから取り出す。検査は以下の手順で行う。

- １つ目の区間に現れる命令番号をファイルＡに保存する。ただし、同じ番号が複数回現れるときは、その番号を１つだけ保存する。
- ２つ目の区間に現れる命令番号をファイルＢに保存する。１つ目の区間と同様に、同じ番号が複数回現れるときは、その番号を１つだけ保存する。
- ファイルＡに保存されたすべての番号と、ファイルＢに保存されたすべての番号を、ファイルＣに１回ずつ保存する。したがって、同じ命令番号がファイルＡとＢの両方に保存されている場合、その番号はファイルＣに２回現れる。
- ファイルＣに、1 からMまでの各番号がちょうど 1 回現れ、それ以外の番号が現れなければ「Yes」とする。そうでなければ「No」とする。

課題

命令ログが与えられたとき、検査を何度か行い、各検査の結果を出力するプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

$N\ M\ L\ Q$
$x_1\ x_2\ \ldots\ x_L$
$a_1\ b_1\ c_1\ d_1$
$a_2\ b_2\ c_2\ d_2$
$\vdots$
$a_Q\ b_Q\ c_Q\ d_Q$

１行目に命令の数$N(1 \leq N \leq 100{,}000 = 10^5)$、標準命令の数$M(1 \leq M \leq N)$、命令ログの長さ$L(1 \leq L \leq 100{,}000 = 10^5)$、検査の回数$Q(1 \leq Q \leq 100{,}000 = 10^5)$ が与えられる。続く 1 行に、命令ログに含まれるi番目の命令の番号$x_i(1 \leq x_i \leq N)$ が与えられる。

続くQ行に、i番目の検査を表す４つの整数a_i, b_i, c_i, d_iが与えられる。１つ目の区間はa_iからb_iまで$(1 \leq a_i \leq b_i \leq L)$、２つ目の区間は$c_i$から$d_i$まで$(1 \leq c_i \leq d_i \leq L)$である。

時間制限

入力に対して、実行時間が１秒を超えてはならない。

出力

各検査について、その結果「Yes」または「No」を１行に出力する。

入出力例

入力例１	出力例１
5 5 5 3 1 2 3 4 5 1 3 4 5 1 2 2 5 2 4 3 5	Yes No No

入力例２	出力例２
6 4 8 4 1 3 1 5 4 2 3 6 1 1 5 7 1 3 5 6 1 3 5 7 1 4 5 7	Yes Yes No No

問題１２　コボウ氏の単語帳　（２０点）

コボウ氏は、新しい単語を考えるのを趣味にしている。最近、コボウ氏は新しい単語帳を購入した。この単語帳には、１ページに１つ単語を書くことができる。はじめにコボウ氏は、思いついた単語を単語帳の１ページ目に書いた。この単語帳を使って、次のような方法で単語を作ってみることにした。

1. 単語帳のpページ目を開く。そのページに書かれている単語をWとする。
2. 単語Wの先頭m文字からなる文字列をsとする（$m = 0$のときsは空の文字列となる）。sの後ろに新たな文字列tをつなげて、単語stを作る。
3. 現在までに単語帳の１ページ目からNページ目まで単語が書いてあるものとすると、作った単語を単語帳の$N + 1$ページ目に書く。

また、コボウ氏はこうして作った単語帳のページをときどき開いて、単語を眺めて楽しんでいる。コボウ氏は眺める単語が書かれているページを以下の方法で開く。

1. 単語が書いてあるページの中からpページ目を開く。そのページに書いてある単語をWとする。
2. 単語Wの先頭m文字からなる文字列をsとする（$m = 0$のときsは空の文字列となる）。
3. 単語帳に書かれた単語の中から先頭がsである単語をすべて抜き出す（複数のページに同じ単語が書かれているときは、それらすべてを抜き出す）。抜き出した単語を辞書式順序（英和辞書で単語が並んでいる順番）に並べ、先頭からk番目の単語Tを眺めることにする。
4. 単語Tが書かれたページが複数あるときは、その中で最も前のページを開く。

たとえば、先頭が“ab”の単語“abc”, “abcd”, “abka”, “abcd”がそれぞれ単語帳の３、７、１５、２８ページに書かれているとする。これらを辞書式順序で並べると“abc”, “abcd”, “abcd”, “abka”となる。先頭から３番目のものを選ぶとすると“abcd”が選ばれる。“abcd”は７ページと２８ページに書かれているので、コボウ氏は７ページ目を開いて単語“abcd”を眺めて楽しむ。

※ 辞書式順序とは、２つの文字列を左から順に比較し、最初に異なる文字が現れた位置で、アルファベット順で前の文字を持つ方を先とする順序である。一方の文字列がもう一方の文字列の先頭部分と一致する場合は、短い文字列を先とする。

課題

コボウ氏が単語を書いたり眺めたりする動作の情報が与えられているとき、コボウ氏が眺めるページの番号を順に出力するプログラムを作成せよ。

入力

入力は以下の形式で与えられる。

V
Q
$action_1$
$action_2$
⋮
$action_Q$

１行目に、単語帳の１ページ目に書く単語を表す英小文字からなる長さ1以上$200{,}000 = 2 \times 10^5$以下の文字列$V$が与えられる。続く１行に、コボウ氏の動作の数$Q(1 \leq Q \leq 200{,}000 = 2 \times 10^5)$が与えられる。続く$Q$行にコボウ氏の$i$番目の動作の情報$action_i$が与えられる。$action_i$は以下の形式で与えられる。

```
1 p m t
```

または

```
2 p m k
```

最初の数字が 1 のときは単語帳に単語を書く動作、2 のときは単語帳のページを開いて単語を眺める動作を表す。ただし、$1 \leq p \leq N$、$0 \leq m \leq |W|$、$1 \leq k \leq C$、tは英小文字からなる長さ1以上$200{,}000 = 2 \times 10^5$以下の文字列とする。ここで、$N$は動作の直前の時点での単語帳に書かれた単語の数、$|W|$は$p$ページ目に書かれている単語$W$の長さ、$C$は動作の直前の時点で単語帳に書かれている単語のうち、先頭がs（Wの先頭からm文字目までの文字列）である単語が書かれたページ数である。

なお、以下の条件が満たされるものとする。

- 入力に現れるすべてのtの長さの和にVの長さを加えた値は$200{,}000 = 2 \times 10^5$を超えない。
- 単語帳のページを開いて単語を眺める動作は１つ以上与えられる。

時間制限

入力に対して、実行時間が３秒を超えてはならない。

出力

単語帳のページを開いて単語を眺める動作ごとに、コボウ氏が眺める単語が書かれた最も前のページの番号を１行に出力する。

入出力例

入力例	出力例
abc 5 1 1 1 dfp 2 1 1 2 1 2 0 abc 2 1 1 2 2 1 0 3	2 1 2

はじめにコボウ氏は、“abc”を１ページ目に書く。その後、以下の５つの動作を行う。

- １ページ目の単語“abc”の先頭１文字からなる文字列“a”と“dfp”をつなげて作った単語“adfp”を２ページ目に書く。
- １ページ目の単語“abc”の先頭１文字からなる文字列“a”を先頭に持つ単語は“abc”と“adfp”である。これらを辞書式順序で並べると、先頭から２番目は２ページ目に書かれた“adfp”なので、2を出力する。
- ２ページ目の単語“adfp”の先頭０文字からなる文字列“”（空の文字列）と“abc”をつなげて作った単語“abc”を３ページ目に書く。
- １ページ目の単語“abc”の先頭１文字からなる文字列“a”を先頭に持つ単語は、２ページ目に書かれた“adfp”と１ページ目と３ページ目に書かれた２つの“abc”である。これらを辞書式順序で並べた“abc”, “abc”, “adfp”の先頭から２番目は“abc”である。“abc”は１ページ目と３ページ目に書かれているので、1を出力する。
- １ページ目の単語“abc”の先頭０文字からなる文字列“”（空の文字列）を先頭に持つ単語は、２ページ目に書かれた“adfp”と１ページ目と３ページ目に書かれた２つの“abc”である。これらを辞書式順序で並べた“abc”, “abc”, “adfp”の先頭から３番目は２ページ目に書かれた“adfp”なので、2を出力する。