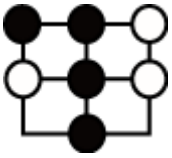
問題01　三目並べ（30点）

この問題は、審査委員特別賞の対象となります。



３×３の盤面の三目並べを考えましょう。三目並べは2人で行う対戦ゲームです。先攻後攻を決め、一人は黒石、一人は白石を打ちます。盤面に石を交互に一つずつ置いていき、縦横斜めのいずれかの方向に先に3つ自分の石を並べた人が勝ちとなります。

盤面の情報を入力とし、勝敗の判定を行い、黒が勝ちの場合は「b」、白が勝ちの場合は「w」、どちらもそろっていない場合は「NA」と出力するプログラムを作成してください。

盤面の情報は、３行３列の文字列で構成されます。「b」は黒石、「w」は白石、「+」（半角プラス）は何も置かれていない状況を表します。ただし、同時に黒が3つ、白が3つ並ぶことはありません。

入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロひとつの行で示されます。

各データセットは以下のとおりです。

１行目　盤面の第１行目の情報（半角文字列）
２行目　盤面の第２行目の情報（半角文字列）
３行目　盤面の第３行目の情報（半角文字列）

出力

入力データセット毎に、「b」、「w」、「NA」のいずれかを出力します。

入力例	出力例
bbw wbw +b+ bwb wbw wbw 0	b NA

問題02　鶴ヶ城 （20点）

会津若松市のシンボルである鶴ヶ城は、蒲生氏郷が本格的な天守閣を築城し、「鶴ヶ城」と名付けました。天守閣からは会津盆地が一望できます。また、晴れた日には、白虎隊で有名な飯盛山の山頂から鶴ヶ城を見ることができます。

会津若松市の今後の広報活動の参考にするため、鶴ヶ城に訪れた来場者について、年代調査をすることにしました。来場者の年齢を入力とし、下記の年齢区分別に人数を出力するプログラムを作成してください。ただし、来場者の人数は1,000,000人未満とします。

区分	年齢
１０歳未満	０　～　９
１０代	１０　～　１９
２０代	２０　～　２９
３０代	３０　～　３９
４０代	４０　～　４９
５０代	５０　～　５９
６０歳以上	６０　～

入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロひとつの行で示されます。

各データセットは以下のとおりです。

1行目　来場者の人数 n（整数）
2行目　1人目の来場者の年齢（整数）
3行目　2人目の来場者の年齢（整数）
　　：
　　：
n+1行目　n人目の来場者の年齢（整数）

出力

入力データセット毎に以下の形式で出力します。

1行目　10歳未満の人数（整数）
2行目　10代の人数（整数）
3行目　20代の人数（整数）
4行目　30代の人数（整数）
5行目　40代の人数（整数）
6行目　50代の人数（整数）
7行目　60歳以上の人数（整数）

入力例	出力例
8 71 34 65 11 41 39 6 5 4 67 81 78 65 0	2 1 0 2 1 0 2 0 0 0 0 0 0 4

問題03 ゴールドバッハ予想 （20点）

　ゴールドバッハ予想とは「6以上のどんな偶数も、2つの素数※1の和として表わすことができる」というものです。

たとえば、偶数の12や18は
12=5+7，18=5+13=7+11
などと表せます。
　和が134となる2つの素数の組み合せをすべて書き出すと、以下の通りとなります。
　134　=3+131　=7+127　=31+103　=37+97　=61+73　=67+67
　　=131+3　=127+7　=103+31　=97+37　=73+61
　与えられた数が大きくなると、いくらでも素数の組み合せが見つかるような気がします。しかし、現代数学をもってしてもこの予想を証明することも、反例を作ることもできません※2。ちょっと不思議な感じがしますね。
　そこで、ゴールドバッハ予想を鑑賞するために、偶数 n を入力とし、和がnとなる2つの素数の組み合せの数を求めるプログラムを作成してください。
　和がnとなる素数の組み合せの数とは n = p+q かつ p ≦ q であるような正の素数 p、q の組み合せの数です。上の例からわかるように和が 134 となる素数の組み合せは6個です。ただし、n は 6 以上 100,000 以下の偶数とします。

入力

　複数のデータセットの並びが入力として与えられます。入力の終わりはゼロひとつの行で示されます。
　各データセットは以下のとおりです。

　1行目　偶数 n

出力

　入力データセット毎に素数の組み合せの数を出力します。

入力例	出力例
134 4330 34808 98792 0	6 72 274 607

※1 素数とは、１または自分自身以外に約数を持たない整数のことである。なお、１は素数ではない。
※2 2007年2月現在、5×10^{17} までの全ての偶数について成り立つことが確かめられている。(Wikipedia)

問題04　会津地鶏 （20点）

2008年4月に、会津若松市は長さ20m85cmのやきとり作りに挑戦して成功しました。このとき使われた鶏肉が、会津特産の会津地鶏です。会津地鶏はとてもおいしいのですが、飼育がむずかしいので生産量が少なく、値段が高いのが難点です。現在、生産拡大のための努力が進められているので、そのうちみなさんの住む町でも目にする日が来るかもしれません。

今日は一郎君の家に、遠くから親戚が遊びに来ます。お母さんは鶏肉の水炊きを作って親戚をもてなすことにしました。近くのお肉屋さんでは会津地鶏とふつうの鶏肉の２種類の鶏肉を売っています。お母さんは、以下のような指示を一郎君に与えて、お肉屋さんに行って鶏肉を買ってくるように頼みました。

- 鶏肉が足りなくなると困るので、決められた量以上の鶏肉を買う。
- 予算の許す範囲で会津地鶏をできるだけ多く買う(会津地鶏は必ず買うこと)。
- 会津地鶏を買った残りでふつうの鶏肉をできるだけ多く買う(予算が足りなければ買わない)。

一郎君がお肉屋さんに行くと、会津地鶏が品薄のため一人が買える量を制限していました。一郎君が、お母さんから与えられたすべての指示を守って買い物をするとき、一郎君が買う会津地鶏と普通の鶏肉の量はそれぞれ何グラムになるでしょう？

買うべき鶏肉の量の下限 q1、予算 b、このお肉屋さんでの会津地鶏100グラムの値段 c1、ふつうの鶏肉100グラムの値段 c2、会津地鶏を一人が買える量の上限 q2 を入力とし、一郎君が買う会津地鶏とふつうの鶏肉の量を100グラム単位で出力するプログラムを作成してください。ただし、このお肉屋さんではお母さんの指示通りに買えない場合には、「NA」と出力してください。

鶏肉の量を表すデータ q1 と q2 は100グラム単位で指定され、c1 > c2とします。また、b、c1、c2、q1、q2 は 1 以上 1,000,000 以下の整数とします。

入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロひとつの行で示されます。

各データセットは以下のとおりです。

1行目　q1 b c1 c2 q2（それぞれ整数；半角空白区切り）

出力

入力データセット毎に一郎君が購入する会津地鶏の量とふつうの鶏肉の量（半角空白区切り）、または「NA」を出力します。

入力例	出力例
48 9297 240 126 32 20 3010 157 141 7 30 117002 5680 962 15 8 1673 1712 190 22 64 8478 87 54 307 23 5477 117 92 12 50 7558 1396 187 17 279 88677 4522 514 14 0	28 20 7 13 15 33 NA 97 0 12 44 NA NA

パソコン甲子園２００８本選問題

問題05　投石おみくじ（20点）

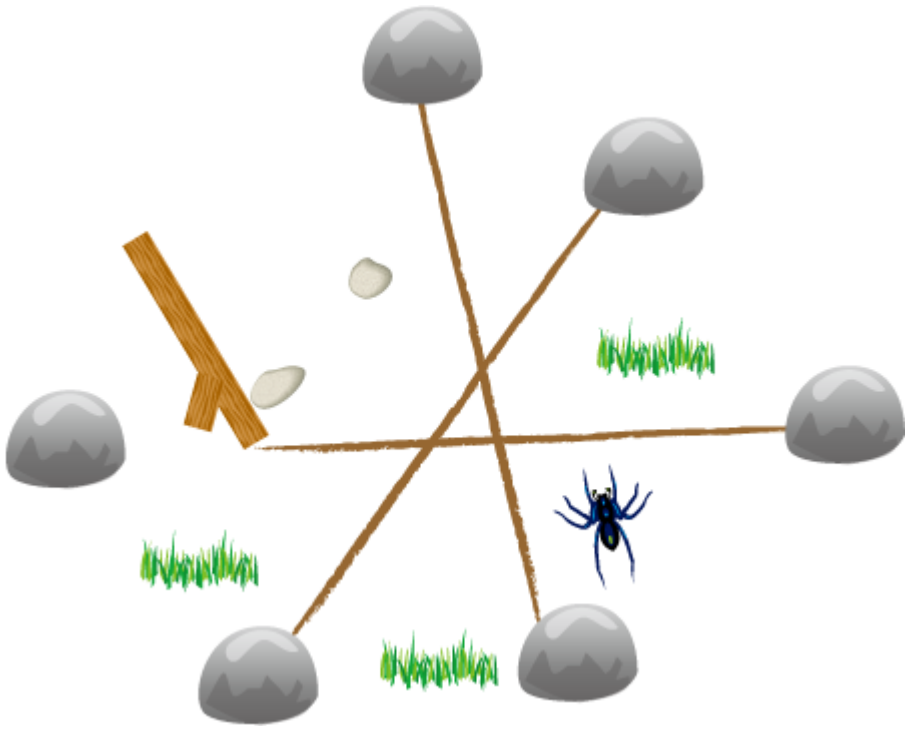
「自分の道は自分で切り開く」をモットーに、ある神社が自分自身の手で運勢を決めるおみくじを作りました。そのおみくじを引く人にはまず６つの石を投げてもらい、その投げた石の一つ目と二つ目を結ぶ 線分、三つ目と四つ目を結ぶ 線分、五つ目と六つ目を結ぶ 線分の３本の線分の交点を頂点とする 三角形の面積から運勢を決めるというものです。各運勢と三角形の面積との関係は以下の図に示してあります。

線分の交点を頂点とする三角形の面積	運勢
1,900,000以上	大吉（dai-kichi）
1,000,000以上1,900,000未満	中吉（chu-kichi）
100,000以上1,000,000未満	吉（kichi）
0より大きく 100,000未満	小吉（syo-kichi）
三角形なし	凶（kyo）

しかし今の三角形の面積の大きさの判定は神主さんが手計算でやっているので正確とはいえず、時間もかかってしまいます。そこで近所に住む優秀なプログラマであるあなたは、一刻でも早くプログラムを書いて神主さんを助けてあげることにしました。

3本の線分の情報を入力とし、線分の交点を頂点とする 三角形の面積から運勢を出力するプログラムを作成してください。線分の情報は始点の座標（x1,y1）と、終点の座標（x2,y2）が与えられ、始点と終点の座標は必ず異なることとします。また、2つ以上の線分が同一直線上にある場合、交点を持たない2つの線分がある場合、3つの線分が１点で交わる場合は、「三角形なし」となります。

x1、y1、x2、y2 は -1,000以上 1,000以下の整数とします。



入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロ4つの行で示されます。
各データセットは以下のとおりです。

1行目　第1の線分の情報　x1 y1 x2 y2（それぞれ整数　；半角空白区切り）
2行目　第2の線分の情報
3行目　第3の線分の情報

出力

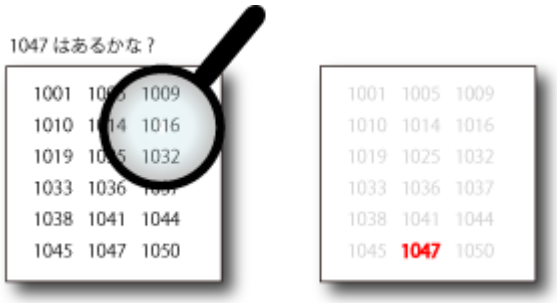
入力データセット毎に、おみくじの結果を出力します。

入力例	出力例
-3 -2 9 6 3 -2 7 6 -1 0 5 0 2 2 -1 -1 0 1 2 1 -3 -1 3 1 0 0 0 0	syo-kichi kyo

パソコン甲子園２００８本選問題

問題06　探索（40点）

「探索」とは、たくさんの情報の中から望みの情報を得る操作のことです。身近な例では、合格発表の時の「たくさんの受験番号から自分の受験番号を見つける」ことや、電話帳から「会津太郎さんの電話番号を見つける」ときなどがあります。この探索という操作はコンピュータの分野でも広く用いられています。

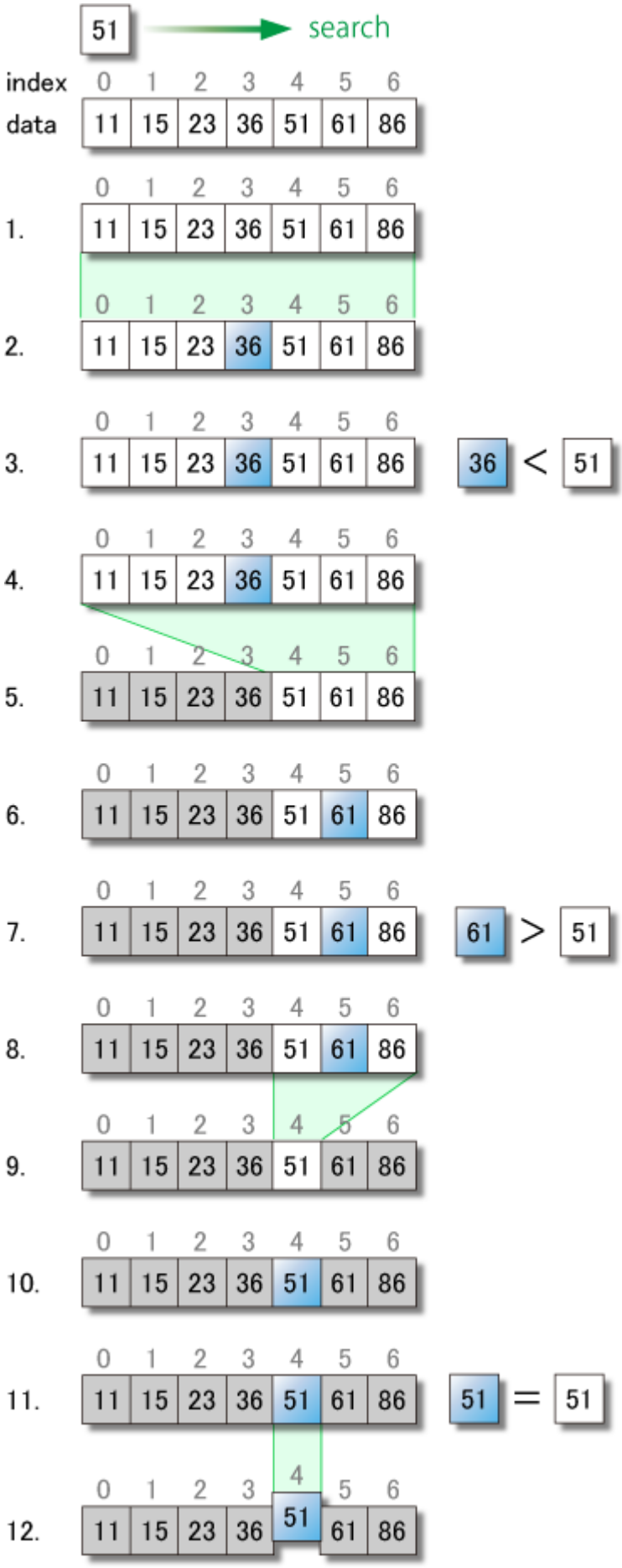


探索の方法は沢山あります。探索の対象となるデータが、小さい順（または大きい順）に並べられている場合に使うことができる探索の方法を考えます。

小さい順（または大きい順）に並べられているデータ列の中央に位置する値と目的の値との大小関係を用いて、中央に位置する値から前半部分を探索範囲にするか後半部分を探索範囲にするかを決めることで、探索の範囲を絞っていく方法があります。手順は以下のようになります。

① データ列の全体を探索の範囲とします。
② 探索の範囲の中央に位置する値を調べます。
③ 目的の値と中央に位置する値が一致すれば探索を終了します。
④ 目的の値が中央に位置する値よりも小さければ前半部分を探索の範囲とし、大きければ後半部分を探索の範囲として② へ戻ります。

以下は上述した探索の方法の例です。この例での目的の値は51です。それぞれのデータには番号（index）が振られており、この番号は0から始まっています。



ステップ１：
最初は番号0～6全体を探索の範囲とします。

ステップ２：探索の範囲の中央に位置する値を調べます。ただし、「中央に位置する値」とは、(左側の番号＋右側の番号)を2で割った番号の位置にある値とします。つまり、この場合、(0 + 6) ÷ 2 を計算し、番号3にある値(36)が中央に位置する値となります。

ステップ３：目的の値(51)と中央に位置する値(36)を比較します。

ステップ４：ステップ３の結果から、目的の値は中央に位置する値より大きいので、後半部分にあたる番号4（中央に位置する値の隣）以降を探索の範囲とします。

同様の手順で探索の範囲の中央に位置する値を調べ、目的の値が中央に位置する値より小さければ前半部分を探索の範囲とし、大きければ後半部分を探索の範囲として、探索の範囲を小さくしていきます。（ステップ２～ステップ４の繰り返し）

目的の値が中央に位置する値と一致するか、探索の範囲がつきてしまった時に探索を終了します。

n個の数値の配列を入力とし、目的の値と中央に位置する値を比較した回数を出力するプログラムを作成してください。ただし、中央に位置する値の番号を計算したとき、割り切れない場合は、小数点以下を切り捨てた値をその番号とすることとします。また、n は 1以上 100以下とし、入力される値は 1以上 100,000以下の整数とします。さらに、与えられるデータ列は小さい順に整列されているものとします。

入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロひとつの行で示されます。
各データセットは以下のとおりです。

1行目　数値の数　n（整数）
2行目　1つ目の数値（整数）
3行目　2つ目の数値
　　：
　　：
n+1行目　n個目の数値
n+2行目　探索する値（整数）

出力

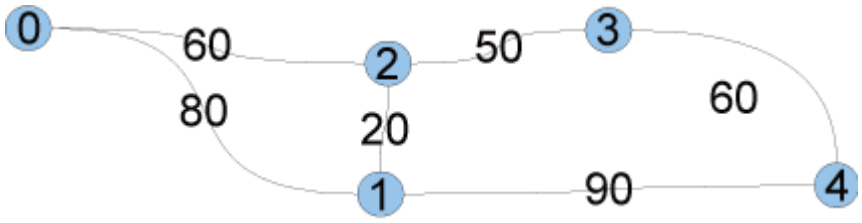
入力データセット毎に探索が終わるまでの比較回数（整数）を出力します。

入力例	出力例
7 11 15 23 36 51 61 86 51 4 1 2 3 5 4 0	3 3

問題07　便利なのはどこ？（40点）

来春卒業するAさんは，就職を機に引越しをすることにしました。就職する会社は，オフィスがいくつかの町にあって、日によって出勤するオフィスが違います。そこでAさんは，どこのオフィスに行くにも時間の短い町に住もうと考えました。

そこであなたは、Aさんを助けるため、住むのに一番便利な町を探すことになりました。



町には0から始まる番号が振られており、町と町の間には道があります。それぞれの道に対して通勤時間が決まっています。Aさんがある町に住んでいる場合に、自分の町のオフィスまでの通勤時間は0とします。このときに全ての町までの通勤時間の総和を考えます。例えば，町と道の配置が上の図のようになっていて、Aさんが町1に住んだ場合には、それぞれの町までの通勤時間は

町0まで　80
町1まで　 0
町2まで　20
町3まで　70
町4まで　90

となり、総和は260となります。

道の数と、全ての道の情報を入力とし、それぞれの町に住んだ場合の通勤時間の総和を計算し、それが最小となる町の番号と、そのときの通勤時間の総和を出力するプログラムを作成してください。ただし、通勤時間の総和が最小となる町が複数ある場合は、一番小さい町の番号及びその時の通勤時間の総和を出力してください。町の総数は10以下、道の総数は45以下とし、全ての道は双方向に移動でき、通勤時間は方向によって変わらないものとします。また、どの町からでもその他全ての町への経路があるものとします。

入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロひとつの行で示されます。
各データセットは以下のとおりです。

1行目　道の数n（整数）
2行目　第1の道の情報 a1 b1 c1（それぞれ整数；半角空白区切り）
　　各記号の意味は以下のとおりです。
　　a1 b1：この道がつないでいる町の番号
　　c1：a1、b1間の通勤時間
3行目　第2の道の情報 a2 b2 c2
　　：
n+1行目　第nの道の情報　an bn cn

出力

入力データセット毎に通勤時間の総和が最小になる町の番号及びその時の通勤時間の総和を出力します。（半角空白区切り）

入力例	出力例
6 0 1 80 1 2 20 0 2 60 2 3 50 3 4 60 1 4 90 2 0 1 1 1 2 1 0	2 240 1 2

パソコン甲子園２００８本選問題

問題08　11パズル（70点）

太郎君は８パズルが大得意で休み時間などにいつも友達に並び替えてもらって遊んでいます。そんなとき、友達から「もっと複雑なパズルは解ける？」と聞かれたのですが、他のパズルはやったことはありません。どうやらその友達は自作で１１パズルを作っていたみたいです。そのパズルは以下のような形をしています。

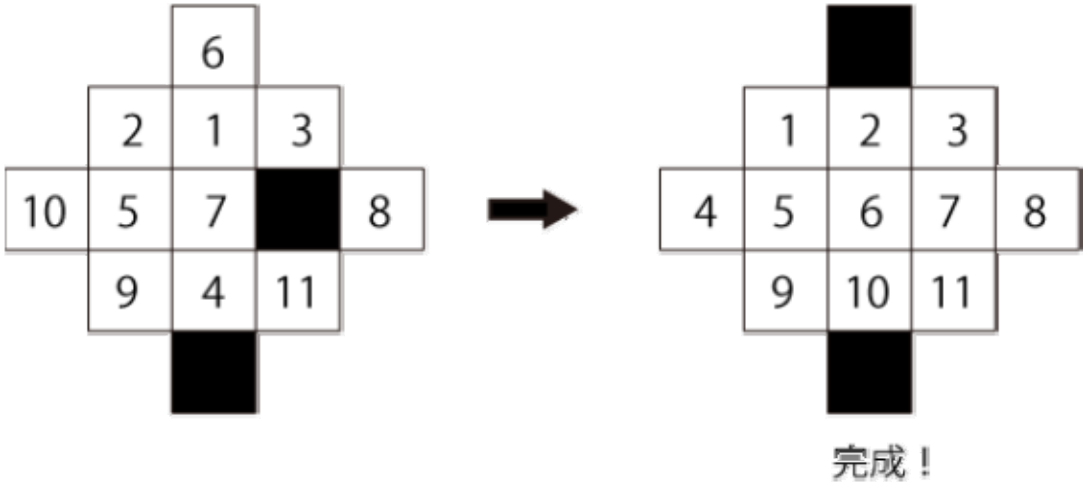


図１　　図２

１１パズルは11枚の正方形のカードと、図1のような形の枠を使って行います。最初に11枚のカードを枠に入れます。すると2カ所の空きスペースができますので、この空きスペースに隣接したカードを動かすことができます。これを繰り返し、カードをきれいに整列して、図2の完成型にすることが１１パズルの目的です。

太郎君はこのパズルに挑戦することにしました。ところが太郎君はこの１１パズルをいとも簡単に解いてしまいました。そこで友達は「動かす数を一番少なくして解いてよ！」と無茶なことを言ってきました。太郎君は答えがわからないので、プログラムのできるあなたに１１パズルを解くときの最小ステップ数を出すプログラムを作成してもらってから挑戦することにしました。このとき、2カ所動かせるところがあるのですが、一つの数字を移動させることを1ステップとして考えることとします。

１１パズルの初期状態を入力とし、１１パズルを解くときの最小ステップ数を出力するプログラムを作成してください。ただし、パズルを解くときの最小ステップ数が20ステップより多くかかってしまう場合は、「NA」と出力してください。パズルの状態は、一行目の情報から順に入力されるものとし、数字の0は空きスペースを表します。例えば、図1の状態を表す入力は以下のようになります。

```
6
2 1 3
10 5 7 0 8
9 4 11
0
```

入力

複数のデータセットの並びが入力として与えられます。入力の終わりは-1ひとつの行で示されます。
各データセットは以下のとおりです。

1行目　パズルの1行目の情報 p1（整数）
2行目　パズルの2行目の情報 p2 p3 p4（それぞれ整数；半角空白区切り）
3行目　パズルの3行目の情報 p5 p6 p7 p8 p9（それぞれ整数；半角空白区切り）
4行目　パズルの4行目の情報 p10 p11 p12（それぞれ整数；半角空白区切り）
5行目　パズルの5行目の情報 p13（整数）

出力

入力データセット毎に、入力で与えられたパズルを解くときの最小ステップ数またはNAを出力します。

入力例	出力例
2 1 0 3 4 5 6 7 8 9 0 11 10 0 1 2 3 4 5 6 7 8 9 10 11 0 0 11 10 9 8 7 6 5 4 3 2 1 0 -1	2 0 NA

パソコン甲子園２００８本選問題

問題09　おおきくなあれ（70点）

植物学者のサトー博士は苗木用の特殊肥料を何種類も発明しました。その肥料を苗木に与えると、瞬く間に苗木の大きさが変わります。
但し、肥料に以下のように副作用があることが判明しました。

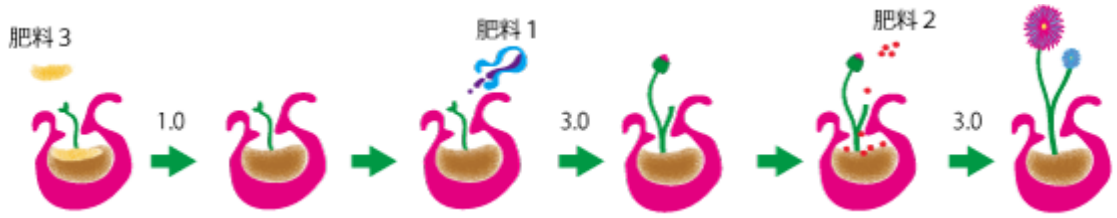
- １回目に与えた肥料だけでは、苗木の大きさが変わりません。
- ２回目以降は、その回に与えた肥料と、その直前に与えた肥料との組み合わせによって苗木に影響を与えます。良い影響を与えると苗木が伸び、悪い影響を与えると苗木が縮んでしまうこともあります。

試しに、サトー博士は３種類の肥料（肥料１、２，３）に対し、ある時点で与えた肥料（今回の肥料）とその直前に与えた肥料（直前の肥料）の組み合わせによる苗木の成長度合いを調べ、以下の「成長度表」を作成しました。

右表の1行目は今回与える肥料の番号で、1列目はその直前に与えた肥料の番号です。他の数字は直前に与えた肥料と今回与える肥料の組み合わせによる苗木の成長度（成長後対成長前の大きさの比率）を示します。成長度 ＞ 1.0 の場合は苗木が伸びる、成長度 ＜ 1.0 の場合は苗木が縮むことを示します。例えば肥料１の後に肥料２を与えると苗木の大きさが３倍になるが、肥料１の後に肥料３を与えると苗木の大きさが半分に縮んでしまいます。

直前の肥料＼今回の肥料	1	2	3
1	1.3	3.0	0.5
2	2.4	2.1	1.0
3	3.0	0.8	1.2

苗木に与える肥料の回数が制限された場合、苗木をできるだけ大きく育てるにはどの肥料をどのような順で与えればよいでしょうか？「成長度表」がその答え教えてくれます。例として上の表にある肥料を3回だけ与える場合、以下のように肥料３→肥料１→肥料２の順にあげると最も苗木が成長します。



1. １回目の肥料（肥料３）では苗木の大きさは変わりません。
2. ２回目の肥料（肥料１）では、表より肥料３後の肥料１での成長度が3.0なので、苗木の大きさは前回の3.0倍になります。
3. ３回目の肥料（肥料２）では、表より肥料１後の肥料２での成長度が3.0なので、苗木の大きさはさらに前回の3.0倍で、最初の3.0×3.0の9.0倍になります。

今度は、サトー博士は発明した n 種類の肥料を全部調べて、上のような「成長度表」を作りあげましたが、非常に大きな表になってしまい、肥料の種類と与える順番を決めるのに大変苦労しています。

そこで博士に代わり、n種類の肥料の組み合わせによる苗木の「成長度表」中の成長度値部分を入力とし、肥料を m 回与えた後の最大の苗木の大きさを求めるプログラムを作成してください。ただし、初めの苗木の大きさを 1 とし、1回目に与える肥料の成長度はどの肥料でも1.0とします。なお、肥料は 1 から n まで番号付けされています。肥料の種類 n は 2以上 100以下の整数、与える回数 m は 1以上 100以下の整数とします。
出力する苗木の大きさは、小数点第3位を四捨五入して、小数点第2位まで求めてください。

入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロふたつの行で示されます。
各データセットは以下のとおりです。

1行目　肥料の種類n 肥料を与える回数 m（整数 整数；半角空白区切り）
2行目　g_{11} g_{12} … g_{1i} … g_{1n} （それぞれ実数；半角空白区切り）
　　但し g_{1i} は肥料 1 の後に肥料 i を与えた時の苗木の成長度
　　:
　　:
n+1行目　g_{n1} g_{n2} … g_{ni} … g_{nn} （それぞれ実数；半角空白区切り）
　　但し g_{ni} は肥料 n の後に肥料 i を与えた時の苗木の成長度

出力

入力データセット毎に、最大の苗木の大きさを出力します。

入力例	出力例
3 3 1.3 3.0 0.5 2.4 2.1 1.0 3.0 0.8 1.2 2 2 1.0 1.0 1.0 1.0 0 0	9.00 1.00

問題10　鶴ヶ駐車場（90点）

街中には駐車場の利用効率を上げるため、立体式やタワー式などの様々な駐車場があります。その中には、ひとつの駐車スペースに図のような「２段式駐車装置」を設置し、２台分の駐車スペースを確保している駐車場もあります。この２段式駐車装置は１台を昇降式のパレット（車を乗せる平らな鉄板）に乗せて上段に駐車させ、もう１台を下段に駐車することができます。

このような２段式駐車装置を用いている駐車場では、上段の車を出し入れするのに、その都度、下段に駐車されている車を出して、退かす必要があるので、必ず管理人さんが駐車している車のカギを預かって、必要に応じて車の出し入れを行います。

鶴ヶ駐車場もこのような２段式駐車装置を設置している駐車場のひとつですが、人手不足のため、車の運転ができない人が管理人になってしまいました。そのため、一度駐車した車はお客さんが戻るまで動かすことができず、上段になった車は下段の車の持ち主が戻ってからでないと車を出すことができない状態になってしまいました。そんな条件付きでも、鶴ヶ駐車場は立地条件が良く、使うお客さんがたくさんいました。次から次へと駐車しに来る車を手際よくさばかなければならない管理人さんを手伝うため、鶴ヶ駐車場のルールを満たすプログラムを作成してください。

- 鶴ヶ駐車場の設備
  - 駐車スペースは１つ以上あり、全て２段式駐車装置が設置されています。
  - 各駐車スペースには１から順に番号が割り振られています。
  - 初めは駐車場に１台も駐車していないものとします。

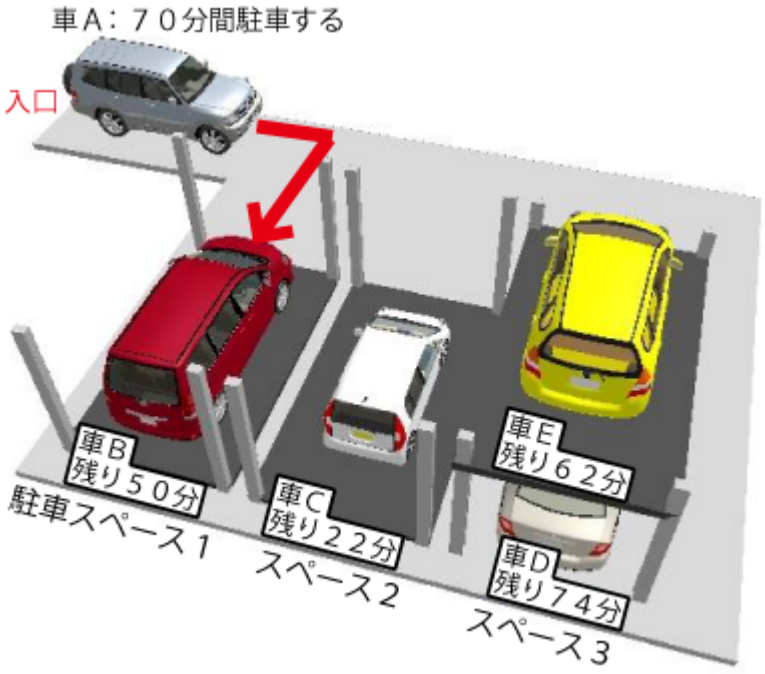
鶴ヶ駐車場は以下のようなルールを採用しています。

- 車を止める時
  - 駐車する車の駐車時間が管理人に知らされます。
  - １台も駐車されていない駐車スペースから先に駐車していきます。
  - １台も駐車されていない駐車スペースがない場合には、空いている駐車スペースに駐車します。
    ただし、そのような駐車スペースが複数あるときは以下の手順で駐車します。
    1. 駐車してある車の残り駐車時間が駐車しようとしている車の駐車時間以上のものがある場合、その差が一番小さい駐車スペースに駐車します。
    2. 駐車してあるどの車の残り駐車時間も駐車しようとしている車の駐車時間未満である場合、その差が一番小さい駐車スペースに駐車します。
  - 満車（空いている駐車スペースがない）の場合、駐車しようとする車は駐車スペースが空くまで順番に待ちます。空いたと同時に、最初に待っていた車から順に駐車します。

    ※各条件において、該当する駐車スペースが複数ある場合は駐車スペース番号の最も小さいところに駐車することとします。また、同時刻に出庫する車がある場合は、出庫する車がすべて出てから駐車を始め、待っている車がある限り、駐車できるだけの車が同時刻に駐車することとします。

- 車が出る時
  - 管理人に知らされた駐車時間を過ぎた車は出庫します。
  - 複数の駐車スペースで同時に駐車時間を過ぎた車があった場合、駐車スペース番号の小さい車から先に出庫します。
  - 上段に駐車した車の駐車時間が過ぎた場合、下段の車が出庫するまで待たなければなりません。上段の車は下段の車が出庫した後、同時刻に出庫します。

下図で鶴ヶ駐車場の駐車方法の例を示します。この例では、駐車スペースの数は3で、車 B ～車 E がすでに駐車してあるとします。そこに駐車時間70分の車 A が来たことを考えます。駐車スペース3にはすでに2台駐車されているので駐車できず、まだ空いている駐車スペース1か駐車スペース2のどちらかに駐車することになります。駐車スペース1に駐車中の車 B の残り駐車時間は50分、駐車スペース2に駐車中の車Cの残り駐車時間は22分で、どちらも車 A の駐車時間より少ないので、車 A の駐車時間との差がより小さい車 B が駐車してある駐車スペース1に駐車します。その結果、先に駐車していた車 B は上段になります。



駐車スペースの数 m、駐車する車の台数 n、各車の駐車時間 t を入力とし、駐車場から出てくる順番に車の整理番号を出力するプログラムを作成してください。ただし、車には入力の順に1からはじまる整数の整理番号が割り振られており、車は10分おきに1台ずつ整理番号順に駐車しにくるものとします。また、m は 1以上 10以下、n は 1以上 100以下、t は 1以上 120以下の整数とします。

入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロふたつの行で示されます。各データセットは以下のとおりです。

1行目　2段式の駐車装置の数m 駐車する車の台数 n（整数 整数；半角空白区切り）
2行目　1台目の車の駐車時間 t1（整数）
3行目　2台目の車の駐車時間 t2（整数）
　　：
n+1行目　n台目の車の駐車時間 tn（整数）

出力

入力データセット毎に駐車場から出てくる順番に車の整理番号を出力します。（半角空白区切り）

入力例	出力例
3 5 90 52 82 84 70 2 4 10 30 40 60 0 0	2 5 1 4 3 1 2 4 3

パソコン甲子園２００８本選問題

問題11　デブンイレブン（90点）

コンビニエンスストア・デブンイレブンは事業を広げるため若松市に一店舗目をオープンしようと考えています。若松市には既に他のたくさんのコンビニエンスストアがあるので、新店舗をオープンする場所が成功の鍵を握ることになりそうです。デブンイレブンは「お客さんは自分の住んでいる地域から最も近いコンビニを利用する」という前提の下、「多くのお客さんが利用するであろう地点」に出店することにしています。

デブンイレブンは、既設の各コンビニがカバーしている範囲を把握するため、若松市の地図を合同な正六角形（以後「ブロック」）を用いて区分しました。このとき、各ブロックには既設のコンビニが多くとも一つだけ入るように区分しました。この地図上で、各ブロックからあるコンビニへ行くのに経由するブロックの数をもとにコンビニとの距離を求めます。各ブロックはこの距離が最も小さいコンビニによってカバーされていると考えます。問題は、この地図をもとに「できるだけ多くのブロックをカバーする既設のコンビニがないブロック」を探すことです。

デブンイレブンのプログラマであるあなたは、ブロック分けされた地図情報と新店舗の候補地の情報から、最も広くカバーできるブロック数を計算するプログラムを開発することになりました。

m×n に区分した地図は図1のように表します。六角形のブロックが横に m 個、縦に n 個並び、それぞれは一つの座標（x, y）で示されます。奇数行の左端はその上の行の左端の左下に、偶数行の左端はその上の行の左端の右下に、それぞれ位置します。例えば、座標（1, 1）は図1の一番左上のブロックを示します。また、座標（1, 2）は座標（1, 1）の右下に位置し、座標（1, 3）は座標（1, 2）の左下に位置します。

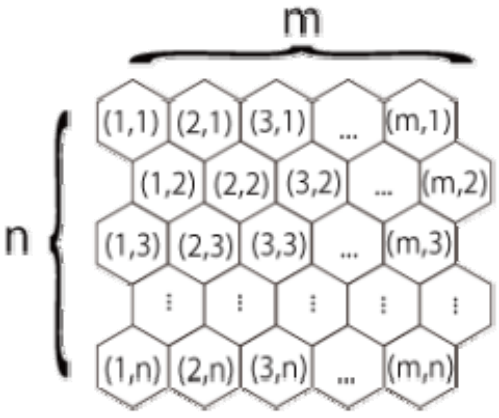


図 1

図2は既設のコンビニが6個あった場合の例で、6×6 に区分されています。各コンビニは１から順に番号が振られています。このとき、各コンビニがカバーするブロックを番号毎に塗り分けると図3のようになります。座標（1，4）や（2，5）のように塗られていないブロックは最も近いコンビニが二つ以上あり、どちらとも判断がつかないブロックになっています。例えば、座標（1，4）のブロックの場合、番号4のコンビニと番号5のコンビニへの距離が等しく、このブロックをカバーしているコンビニはないものとして考えます。番号4のコンビニがカバーしているブロック数は5となります。

図2

図3

ここで、デブンイレブンが座標（1，3）と座標（5，3）を新店舗の候補地として考えているとします。座標（1，3）に店舗を構えた場合、カバーできるブロック数は3となります（図4）。一方、座標（5，3）に店舗を構えた場合にカバーできるブロック数は4となります(図5）。したがって、最も広くカバーできるブロック数は4となります。

図4

図5

地図情報 m×n、既設のコンビニの数 s とその座標（x, y）、候補地の数 t とその座標（p, q）を入力とし、全候補地の中で最も広くカバーできるブロック数を出力するプログラムを作成してください。ただし、m、n はそれぞれ 2以上 100以下の整数、s は 1以上 10以下の整数、t は 1以上 10以下の整数とします。

入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロひとつの行で示されます。

各データセットは以下のとおりです。

1行目　区分した地図の大きさ（横、縦）m n（整数 整数；半角空白区切り）
2行目　既設のコンビニの数 s（整数）
3行目　1個目の既設のコンビニの座標 x_1 y_1（整数 整数；半角空白区切り）
4行目　2個目の既設のコンビニの座標 x_2 y_2（整数 整数；半角空白区切り）
　　：
s+2行目　s個目の既設のコンビニの座標 x_s y_s（整数 整数；半角空白区切り）
s+3行目　新店舗の候補地の数 t（整数）
s+4行目　1個目の候補地の座標 p_1 q_1（整数 整数；半角空白区切り）
s+5行目　2個目の候補地の座標 p_2 q_2（整数 整数；半角空白区切り）
　　：
s+t+3行目　t個目の候補地の座標 p_t q_t（整数 整数；半角空白区切り）

出力

入力データセット毎に、全候補地の中で最も広くカバーできるブロック数を出力します。

入力例	出力例
6 6 6 1 1 6 1 3 2 3 5 1 6 5 6 2 1 3 5 3 6 6 6 3 2 3 5 6 1 1 1 1 6 5 6 2 2 3 5 3 0	4 4

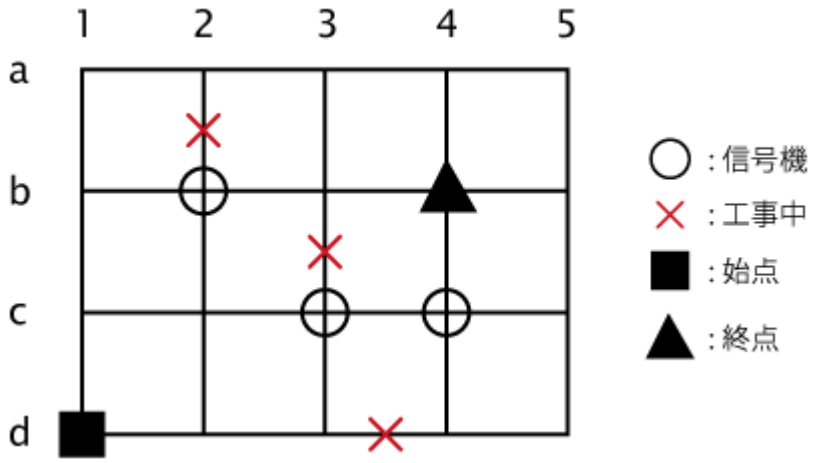
問題12　白虎運送（90点）

白虎運送は会津若松市を代表する運送会社です。昨今の原油価格の高騰は運送会社にも多大なダメージを与え、運送会社各社では、できるだけ少ないガソリンで荷物を運ぶことが課題となっています。

白虎運送では、重い荷物を積んだトラックは、その走りだしに多くのエネルギーを必要とすることに着目しました。トラックが倉庫を出発してから一度も停止することなく目的地まで到達する経路の中で最短時間の経路を通ることで、ガソリンの節約ができると考えました。この仮説が正しいかどうかは不明ですが、白虎運送はあなたが勤務している鶴亀カンパニーに、このような最短経路を計算できるカーナビの開発を依頼しました。あなたはこのカーナビ開発プロジェクトの一員となり、カーナビに埋め込まれる最短時間の経路を計算するプログラムの作成を担当することになりました。

作成するカーナビ用プログラムの仕様は以下のとおりです。

- 市内は、下図のように、東西方向の道路 M本、南北方向の道路 N本 からなる格子で表し、格子の各交点は交差点を表します。
- 交差点のいくつかには東西-南北の方向に信号機が設置されており、一定の周期で青、赤のシグナルが点灯しています。
- 市内の交差点間を結ぶ道路には工事中で通過できない個所がいくつかあります。
- トラックが交差点から交差点へ移動するのに必要な時間は一定ですが、渋滞している道路ではさらに長い時間がかかります。



市内の道路情報、トラックが交差点間を移動するのに必要な時間、信号機がある交差点と各信号機の周期、工事中の道路、渋滞している道路と渋滞度、白虎運送の倉庫（始点）と目的地（終点）の位置を入力とし、始点から終点までの最短の時間を出力するプログラムを作成して下さい。

道路の本数 M、N は 2 以上 20 以下の整数で与えられます。図のように、東西の方向の道路は a、b、c、…とアルファベットで名前が付けられ 、南北の方向の道路は 1、2、3、… と数字で名前が付けられています。市内の交差点は、これらのアルファベットと数字の組み合わせ H-V で指定されます。例えば、市内の最北西の交差点は a-1 で指定されます。各信号は周期 k をもち、k 分ごとに切り替わります。東西が青ならば南北が赤で、南北が青ならば東西が赤です。黄色のシグナルは存

在しません。トラックは二つの交差点を結ぶ道路を移動するのに D 分要しますが、その道路が渋滞している場合はさらに d 分の時間を要します。トラックは道路が工事中の場合は移動できません。また、交差点に到達した時刻に、信号が赤の場合には進入できません。トラックは交差点でのみ、東、西、南、北に方向を変えることができますが、来た方向へは移動（Uターン）できません。道路は2方通行であり、トラックが行き来する時間、工事状況、渋滞度は2方向共通です。

初期状態として、トラックは東を向いていて、トラックが倉庫を出発する瞬間すべての信号機が南北の方向に青に切り替わるものとします。また、トラックは 100分以内で目的地に到達できるものとします。

入力

複数のデータセットの並びが入力として与えられます。入力の終わりはゼロふたつの行で示されます。各データセットは以下のとおりです。

1行目　M N（整数 整数；半角空白区切り）
2行目　D（整数）
3行目　信号機の数 ns（整数）
4行目　1個目の信号機の情報 H-V k（半角英字-整数 整数；半角空白区切り）
　　：
　　：
ns+3行目　ns個目の信号機の情報
ns+4行目　工事中の道路の数 nc（整数）
ns+5行目　1個目の工事中の道路の情報 H1-V1 H2-V2
　　（半角英字-整数 半角英字-整数；半角空白区切り）
　　：
ns+nc+4行目　nc個目の工事中の道路の情報
ns+nc+5行目　渋滞道路の数nj（整数）
ns+nc+6行目　1個目の渋滞道路の情報 H1-V1 H2-V2 d
　　（半角英字-整数 半角英字-整数 整数；半角空白区切り）
　　：
ns+nc+nj+5行目　nj個目の渋滞道路の情報
ns+nc+nj+6行目　始点と終点の情報 H1-V1 H2-V2
　　（半角英字-整数 半角英字-整数；半角空白区切り）

出力

入力データセット毎に最短時間を出力します。

入力例	出力例
4 5 1 3 b-2 3 c-3 2 c-4 1	7 4 8

3 a-2 b-2 b-3 c-3 d-3 d-4 2 b-3 b-4 1 c-1 d-1 1 d-1 b-4 4 5 1 3 b-2 3 c-3 2 c-4 1 3 a-2 b-2 b-3 c-3 d-3 d-4 2 b-3 b-4 1 c-1 d-1 1 d-2 b-4 4 5 1 3 b-2 3 c-3 2 c-4 1 3 a-2 b-2 b-3 c-3 d-3 d-4 2 b-3 b-4 1 c-1 d-1 1 d-3 b-4 0 0	