
問題01 コラッツの問題（20点）

この問題は、審査委員特別賞の対象となります。

正の整数 n に対し、

- n が偶数の時は2で割る。
- n が奇数の時は3倍し、1を足す。

という操作を繰り返すと結果が1になります※。

整数 n を入力とし、結果が1になるまでに繰り返される操作の回数を入力するプログラムを作成してください。整数 n は1以上でかつ上記の計算を繰り返す途中の値が1000000以下となる程度の整数とします。

たとえば、入力として3を受け取った場合は、操作列は

3 → 10 → 5 → 16 → 8 → 4 → 2 → 1

になるので、操作の回数（上の矢印の個数）である7を出力すれば正解になります。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 n （整数）

出力

入力データセットごとに操作の回数（整数）を出力します。

入力例	出力例
3	7
10	6
0	

※任意の正の整数 n に対してこの操作を繰り返すと必ず1になるであろうというのが「コラッツの予想」と呼ばれる問題です。この問題は日本では、「角谷の問題」としても知られている未解決の問題です。コンピュータを利用して非常に大きな数 $3 \times 2^{53} = 27, 021, 597, 764, 222, 976$ 以下については反例がないことが知られていますが、数学的には証明されていません。

問題02 理想の体型（15点）

肥満度を表す指数としてBMI（Body Mass Index）があります。BMIの値は以下の式で計算します。

$$\text{BMI} = \frac{\text{体重 (kg)}}{(\text{身長 (m)})^2}$$

BMIの値が標準値に近いほど「理想の体型」と考えられます。

そこで、BMIの標準値を22とした場合、対象者の情報を入力とし、最も「理想の体型」に近い人の情報を出力するプログラムを作成してください。

ただし、対象者の数 n は1以上1000000以下で各対象者には重複のないように同じく1以上1000000以下の整数値の受付番号 i が振られています。また、身長 h は1cm 以上 200cm 以下のセンチメートル単位で与えられ、体重 w は1kg以上200kg以下のキログラム単位で与えられます。最も「理想の体型」に近い人が二人以上いる場合は、受付番号の小さい方を出力することとします。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 対象者の人数 n （整数）

2行目 第1の対象者の情報 i h w （整数 整数 整数；半角空白区切り）

3行目 第2の対象者の情報

：

：

$n+1$ 行目 第 n の対象者の情報

出力

入力データセットごとに「理想の体型」に最も近い人の受付番号（整数）を出力します。

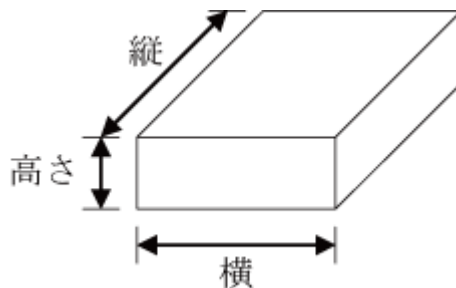
入力例	出力例
6	10003
10001 165 66	2003
10002 178 60	
10003 180 72	
10004 160 65	
10005 185 62	
10006 182 62	
3	
20006 160 65	
2003 180 70	
202 170 75	
0	

問題03 宅配料金 (15点)

ある宅配業者の宅配料金は大きさと重量で下表のように料金が設定されています。大きさは三辺（縦、横、高さ）の和です。例えば大きさが120cmで、重さが15kg以内の荷物はDサイズ（1,200円）となります。大きさが120cm以内でも、重さが15kgを超え、20kg以内ならばEサイズとなります。そこで、1日に持ち込まれた荷物の情報を入力とし、料金の総計を出力するプログラムを作成してください。なお、Fサイズを超えた荷物は対象外とし料金の総計には含みません。ただし、荷物の個数 n は1以上1000000以下とし、入力される縦の長さ (cm) x 、横の長さ (cm) y 、高さ (cm) h 、重さ (kg) w はそれぞれ1以上200以下の整数とします。

料金表

	A サイズ	B サイズ	C サイズ	D サイズ	E サイズ	F サイズ
大きさ	60cm以内	80cm以内	100cm以内	120cm以内	140cm以内	160cm以内
重さ	2kg以内	5kg以内	10kg以内	15kg以内	20kg以内	25kg以内
料金	600 円	800 円	1000 円	1200 円	1400 円	1600 円



プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 荷物の個数 n (整数)

2行目 第1の荷物の情報 x y h w (整数 整数 整数 整数 ; 半角空白区切り)

3行目 第2の荷物の情報

:

:

$n+1$ 行目 第 n の荷物の情報

出力

入力データセットごとに荷物の料金の総計 (整数) を出力します。

入力例	出力例
2	800
50 25 5 5	3800
80 60 10 30	
3	
10 15 25 24	
5 8 12 5	
30 30 30 18	
0	

問題04 体育祭（15点）

秋の体育祭が行われます。種目は徒競走、ボール運び、障害物競走、リレーの4種目です。参加チームはnチームで、この4種目の合計タイムが最も小さいチームが「優勝」、次に小さいチームが「準優勝」、そして、最下位より2番目のチームを「ブービー賞」として表彰したいと思います。

各チームの成績を入力として、「優勝」、「準優勝」、「ブービー賞」のチームを出力するプログラムを作成してください。

ただし、各チームには1以上1000000以下のチーム番号が与えられており、チーム数は4以上1000000以下、タイムを表す分と秒はともに0以上59以下の整数とします。また、合計タイムが複数のチームで同じになるような入力はないものとします。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 対象となるチームの数n（整数）

2行目 第1のチームの情報 c1 m1 s1 m2 s2 m3 s3 m4 s4（整数 整数 …；半角空白区切り）

各記号の意味は以下の通りです。

c1 : チーム番号

m1, s1 : それぞれ徒競走のタイムの分と秒

m2, s2 : それぞれボール運びのタイムの分と秒

m3, s3 : それぞれ障害物競走のタイムの分と秒

m4, s4 : それぞれリレーのタイムの分と秒

3行目 第2のチームの情報

:

:

n+1行目 第nのチームの情報

出力

入力データセットごとに以下の形式で出力します。

1行目 優勝のチーム番号（整数）

2行目 準優勝のチーム番号（整数）

3行目 ブービー賞のチーム番号（整数）

入力例	出力例
8	54092
34001 3 20 3 8 6 27 2 25	34001
20941 3 5 2 41 7 19 2 42	10001
90585 4 8 3 12 6 46 2 34	1
92201 3 28 2 47 6 37 2 58	3
10001 3 50 2 42 7 12 2 54	2
63812 4 11 3 11 6 53 2 22	
54092 3 33 2 54 6 18 2 19	
25012 3 44 2 58 6 45 2 46	
4	
1 3 23 1 23 1 34 4 44	
2 5 12 2 12 3 41 2 29	
3 5 24 1 24 2 0 3 35	
4 4 49 2 22 4 41 4 23	
0	

問題05 ハミング数 (15点)

1に2, 3, 5を何回か(0回以上)かけ算して得られる数をハミング数(Hamming numbers)と呼びます。例えば、

- 1
- $1 \times 2 \times 2 = 4$
- $1 \times 2 \times 2 \times 3 \times 5 \times 5 = 300$

などはハミング数ですが、11, 13, 14などはハミング数ではありません。

ハミング数はどれも60のべき乗を割り切る(たとえば54は $60^3 = 21600$ を割り切る)ので、時刻など60進法の計算には都合の良い数として昔から知られていました。

また、楽器の調律に用いる音階の一つである純正律では、音の周波数の比が 24, 27, 30, 32, 36, 40, 45, 48というハミング数からなる数列になります。

整数 m , n を入力とし、 m 以上 n 以下のハミング数の個数を入力するプログラムを作成してください。ただし、 m , n は1以上1000000以下とし、 m は n 以下とします。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 m n (整数 整数 ; 半角空白区切り)

出力

入力データセットごとに m 以上 n 以下のハミング数の個数を入力します。

入力例	出力例
3 8	5
1 27	17
1 86	31
0	

問題06 有料道路料金（15点）

20XX年に喜多方市熱塩加納町から南会津町までの6区間、総距離58kmの会津中央道路が完成し開通する予定です。開通後、半年間は利用促進のため17時30分～19時30分までに出発ICか到着ICを通過し、なおかつ走行距離が40km以下の車に対する通行料金は半額になります。ただし料金は50円単位とし、端数は切り上げます。下記の表は料金と距離の一覧表です。例えば喜多方（2）から会津若松（4）までは料金が450円、距離が12kmとなります。半額時間帯であれば250円となります。出発IC、出発IC通過時刻、到着IC、到着IC通過時刻を入力とし、料金を計算して出力するプログラムを作成してください。ただし、入力される時刻は24時間表記の値であり、時間を表す h は0以上23以下の整数、分を表す m は0以上59以下の整数とし、ともに2桁で表示します。すなわち、値が10未満の場合は 01 のように一桁目を0で埋めて表記します。なお、17時30分および19時30分ちょうどに通過した場合も半額時間帯に含めます。

料金表

IC名 IC番号	熱塩加納 1	喜多方 2	湯川 3	会津若松 4	会津美里 5	昭和 6	南会津 7
1		300円	500円	600円	700円	1350円	1650円
2	6km		350円	450円	600円	1150円	1500円
3	13km	7km		250円	400円	1000円	1350円
4	18km	12km	5km		250円	850円	1300円
5	23km	17km	10km	5km		600円	1150円
6	43km	37km	30km	25km	20km		500円
7	58km	52km	45km	40km	35km	15km	

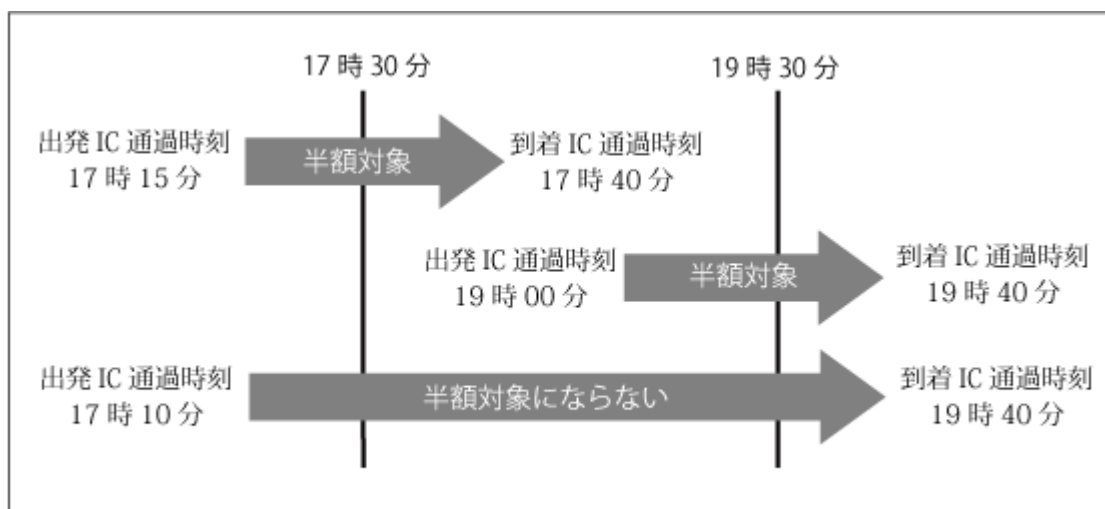


図1 半額時間帯の例

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 出発IC番号（整数）
- 2行目 出発IC通過時刻 h m（整数 整数；半角空白区切り）
- 3行目 到着IC番号（整数）
- 4行目 到着IC通過時刻 h m（整数 整数；半角空白区切り）

出力

入力データセットごとに通行料金（整数）を出力します。

入力例	出力例
2	250
17 25	1300
4	
17 45	
4	
17 25	
7	
19 35	
0	

問題07 おはじき取り (15点)

一郎君と次郎君の兄弟は家でよくおはじき取りをして遊びます。

おはじき取りは、一カ所に積まれた複数のおはじきを二人が交互にとっていくゲームです。一度に1~4個のおはじきを好きな数だけ順に取り、相手に最後の1個を取らせた方が勝ちになります。二人はいつも 32 個のおはじきを使い、兄である一郎君の番からゲームを始めます。

これまでに何度も戦っている二人ですが、次郎君は兄の一郎君にどうしても勝つことができません。それもそのはず、一郎君はこのゲームの必勝法を知っているからです。

一郎君は、残りのおはじきの数を n とすると、必ず

$$(n - 1) \% 5$$

個のおはじきを取ります。ここで $a \% b$ とは、 a を b で割った余りを示します。

一方、次郎君は、残りのおはじきの数にかかわらず、ゲームのはじめに各回で取るおはじきの数を数列として決めてしまうのです。

例えば、次郎君が決めた数列が

3 1 4 2

であるならば、彼の取るおはじきの数は順に $3 \rightarrow 1 \rightarrow 4 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 4 \rightarrow \dots$ となります (取ると決めた数が、おはじきの残りの数以上になった場合は、残りのおはじき全てを取ります)。

なんど負けてもやり方を変えようとしない頑固な次郎君の将来が心配になったお母さんは、時には融通を利かせることも大切であることを説きたいと考えていました。そこで、あなたはお母さんの気持ちを酌んで、次郎君がいかなる数列を選んだとしても一郎君には勝てないということを示すために、ゲームをシュミレートするプログラムを書くことにしました。次郎君の考えた数列 a を入力とし、一郎君と次郎君が順次おはじきを取った後の残りのおはじきの個数を出力するプログラムを作成してください。

ただし、次郎君の決めた数列の長さ i は1以上25以下の整数とし、数列の要素の値は1以上4以下の整数とします。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 次郎君の決めた数列の長さ i

2行目 数列の情報 $a_1 a_2 \dots a_i$ (整数 整数 ...;半角空白区切り)

各記号の意味は以下の通りです。

a_1 : 数列の1番目の数

a_2 : 数列の2番目の数

:

a_i : 数列の*i*番目の数

出力

入力データセットごとに、ゲームの各回でのおはじきが取られた直後のおはじきの数（整数）を出力します。

入力例	出力例
4	31
3 1 4 2	28
3	26
4 3 2	25
0	21
	17
	16
	14
	11
	8
	6
	5
	1
	0
	31
	27
	26
	23
	21
	19
	16
	12
	11
	8
	6
	4
	1
	0

問題08 宝くじ (15点)

ある国の王様は素数とギャンブルが大好きです。この国の通貨の単位はプライムとといいます。パソコン甲子園2007実施の11月におけるプライムのクロス円レートは¥9973とちょうど素数になっていたのも、王様は大喜びです。この国では $1/101$ プライムを1サブプライムとする補助貨幣が使われています。この補助貨幣と2007年、アメリカ合衆国に端を発したサブプライム問題とは全く関係がありません。この国の政府では、国家財政の安定、国民の娯楽、王様の趣味を同時に満足させることを目的に宝くじ振興公社を設立し、宝くじ事業を始めることにしました。素数が大好きな王様は、素数が話題になればそれだけで満足して、賞金をあげたくになります。そこで振興公社は次のような宝くじを考えました。

- くじには0からMPまでの一連の番号がつけられている。MPはこの国で知られている最大の素数で、毎月一日に官報で告示される。同時に、MP以下のすべての素数も発表される。それ以上大きな素数が発見されても、その月の間は、宝くじ振興公社を含む全ての公的な機関では、一日に発表されたMPを最大の素数として扱う。2007年11月1日にはMP=999983 (1000000以下の最大の素数) が発表された。
- 宝くじの販売価格は1サブプライム。
- 宝くじの抽選では、当たりくじの番号 p と賞金算出のための数 m の組 (p, m) が何組か選ばれる。 p, m はそれぞれ0以上MP以下の整数。
- この国で知られている素数の中で $p-m$ 以上 $p+m$ 以下のものが X 個あるとすると、抽選結果 (p, m) の賞金は、 X プライムとなる。
- 抽選結果 (p, m) の賞金 X プライムは番号 p の宝くじを持っている当選者に支払われるが、 $X=0$ の場合もあり、この場合には当選者には賞金は支払われない。
- 賞金のうち1プライムは宝くじの売り上げから支出され、 $X-1$ プライムは王様の内廷費から支出される(王様が払ってくれる)。 $X=0$ ならば宝くじの売り上げから1プライムが内廷費に繰り入れられる。(王様に対し支払われる)
- ひとつの番号 p が複数の当たりくじになることもある。この場合にはそれぞれの抽選結果 (p, m) から算出される賞金の合計が当選者に支払われる。

このくじでは当たりくじを買った人は抽選結果から関係する素数を探し、その個数を数えるので、国民の話題に素数がよく出てきて王様はとてご満悦です。宝くじ振興公社としては当たりくじ1本当たり公社負担が1プライム、販売価格が1サブプライムだから、当たりくじの本数 n を、販売した宝くじ101本あたり1本以下となるようにすれば(すなわち、 $n \leq (\text{販売した本数}) / 101$ とすれば)赤字にはなりません。問題は内廷費からの支出額です。

宝くじ振興公社の歴代の総裁は王様のご学友で、みんな王様と同様に素数が大好きでした。当たりくじに対する内廷費からの支出額を決める会議を楽しんでいました。ところが2007年の人事異動でご学友ではないお役人が総裁に就任しました。不幸なことにこの総裁は王様のことは十分に尊敬していますが、数学が苦手な、素数の話を聞くとすぐに鳥肌が立ちます。そこであなたはこの不幸な新総裁のアシスタントとなって、抽選結果を入力として、2007年11月における宝くじ振興公社が王様に請求する賞金の額を出力するプログラムを作成してください。ただし、抽選結果の数 n は1以上1000000以下であり、請求する賞金の額が負になることはありません。

注意

1. この国における素数の定義も日本の学校教育で学習する内容と同じです。即ち、素数とは1と自分自身以外の約数を持たない自然数をいいます（なお、1は素数ではありません）。
2. 我々は1000003が素数であることを知っていますが、この国では2007年11月段階では知られていません。そのため、999963以上1000003以下の範囲にあるこの国で知られている素数は999983と999979の2個しかありません。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 抽選結果の数 n
2行目 第1の抽選結果の情報 p m (整数 整数 ; 半角空白区切り)
3行目 第2の抽選結果の情報
:
:
 $n+1$ 行目 第 n の抽選結果の情報

出力

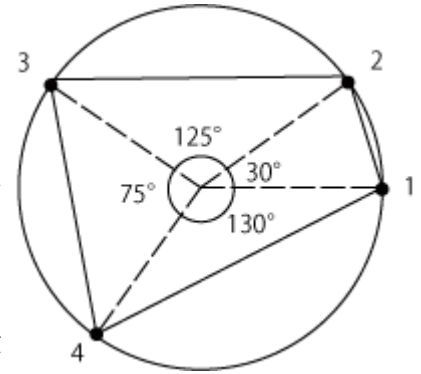
入力データセットごとに王様への請求額をプライム単位（整数）で出力します。

入力例	出力例
4	5
5 0	1
9 1	
3 10	
11 3	
1	
999983 20	
0	

問題09 多角形の面積（15点）

1つの円に内接する2つの多角形の頂点情報を入力とし、それらの面積の大小関係を出力するプログラムを作成してください。

x 角形の各頂点には反時計回りに 1 から x まで番号が振ってあるものとします（図は、 $x = 4$ の場合の例を示しています）。ただし、与えられる多角形は円の中心を内部に含むものとし、頂点 i の位置に関するデータは、頂点 i から頂点 $i+1$ まで反時計回りに計った中心角の角度 v （1 以上 180 未満の整数）で与えられます。また、出力する大小関係については、第1の多角形の面積が大きい場合は 1（半角数字）、第2の多角形の面積が大きい場合は 2（半角数字）、それぞれの面積が等しい場合は 0（半角数字）と出力してください。



プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 第1の多角形の頂点の数 m （整数）
2行目 第1の多角形の頂点1の情報 v_1 （整数）
3行目 第1の多角形の頂点2の情報 v_2
:
 m 行目 第1の多角形の頂点 $m-1$ の情報 v_{m-1}
 $m+1$ 行目 第2の多角形の頂点の数 n （整数）
 $m+2$ 行目 第2の多角形の頂点1の情報 v_1 （整数）
 $m+3$ 行目 第2の多角形の頂点2の情報 v_2
:
 $m+n$ 行目 第2の多角形の頂点 $n-1$ の情報 v_{n-1}

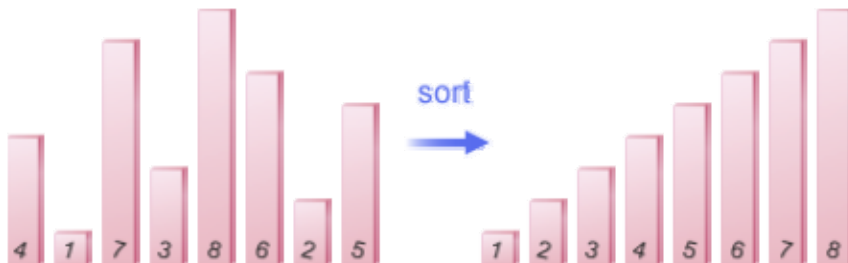
出力

入力データセットごとに大小関係（半角数字）を出力します。

入力例	出力例
4	0
30	1
125	2
75	
130	
4	
30	
125	
75	
130	
5	
30	
125	
75	
65	
65	
4	
30	
125	
75	
130	
4	
30	
125	
75	
130	
6	
30	
50	
50	
25	
75	
130	
0	

問題10 バブルソート (50点)

データを並べ替えるための整列（ソート）アルゴリズムはコンピュータ科学には欠かせない基本的なアルゴリズムです。例えば、下図のように「整数値の配列の要素を昇順に並べ替える」というのが整列です。



多くの整列アルゴリズムが存在しますが、その中でも基本的なアルゴリズムの1つがバブルソートです。例として、与えられた整数値の配列をバブルソートで昇順に並べてみます。

5	3	2	1	4
---	---	---	---	---

バブルソートでは、各計算ステップにおいて、配列を「ソートされた部分」と「ソートされていない部分」に分けて考えます。最初は配列全体がソートされていない部分になります。

5	3	2	1	4
---	---	---	---	---

ソートされていない部分の先頭から、隣同士の要素を比較して（図では緑色の要素）、大きい値が右にくるようにそれらを交換します。二つの値が等しい場合は交換しません。この処理をソートされていない部分（図では白色の要素）の末尾まで繰り返します。最後に、末尾をソートされている部分（図では青色の要素）に追加して1ステップが完了します。

3	5	2	1	4
---	---	---	---	---

3	2	5	1	4
---	---	---	---	---

3	2	1	5	4
---	---	---	---	---

3	2	1	4	5
---	---	---	---	---

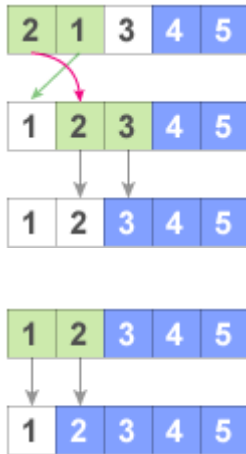
3	2	1	4	5
---	---	---	---	---

このステップをソートされていない部分の長さが1になるまで繰り返します。

2	3	1	4	5
---	---	---	---	---

2	1	3	4	5
---	---	---	---	---

2	1	3	4	5
---	---	---	---	---



このようにソートされていない部分の長さが1になったら、ソートの処理を終了します。

それでは、 n 個の数値の配列を入力とし、数値が配列の先頭から昇順に並ぶように上記のバブルソートの手順で並べ替えを行い、要した配列要素の交換回数を出力するプログラムを作成してください。ただし、 n は1以上100以下とし、入力される数値は1以上1000000以下の整数とします。また、配列の先頭の数値から順に入力されるものとします。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 第1の数値 (整数)
3行目 第2の数値
:
:
:
 $n+1$ 行目 第 n の数値

出力

入力データセットごとにデータ要素の交換回数 (整数) を出力します。

入力例	出力例
5	7
5	0
3	3
2	
1	
4	
6	
1	
2	
3	
4	
5	
6	
3	
3	
2	
1	
0	

問題11 観音堂 (50点)

一郎君の家の裏山には観音堂があります。この観音堂まではふもとから30段の階段があり、一郎君は、毎日のように観音堂まで遊びに行きます。一郎君は階段を1足で3段まで上がることができます。遊んでいるうちに階段の上り方の種類（段の飛ばし方の個数）が非常にたくさんあることに気がつきました。

そこで一日に10種類の上り方をし、すべての上り方を試そうと考えました。このごろ、一郎君はメタボリック症候群の話題が気になっていたのも、ダイエットと、上り方全種類制覇の一挙両得かと思いました。

しかし数学を熟知しているあなたはそんなことでは一郎君の寿命が尽きてしまうことを知っているはずです。そこで、一郎君の計画が実現不可能であることを一郎君に納得させるために、階段の段数 n を入力とし、一日に10種類の上り方をするとして、一郎君がすべての上り方を実行するのに要する年数を出力するプログラムを作成してください。 n は1以上30以下の正の整数とし、一年は365日として計算してください。一日でも必要なら一年とします。365日なら1年であり、366日なら2年となります。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 段数 n (整数)

出力

入力データセットごとに一郎君がすべての上り方を実行するのに必要な年数（整数）を出力します。

入力例	出力例
1	1
10	1
20	34
25	701
0	

【ヒント】

n 段目にいたる一歩手前は $n-1$ 段目か、 $n-2$ 段目か、 $n-3$ 段目のいずれかで、それより下では一足では n 段目には到達できません。 n 段目での上り方を a_n とすると、 a_n は、 a_{n-1} 、 a_{n-2} 、 a_{n-3} の式で表されますね。

問題12 ブラックジャック（50点）

パソコン甲子園で見事優勝したあなたは、賞金でラス・ベガスに遊びに行くことにしました。ラス・ベガスと言えばカジノです。賞金を元手にカジノで一発当てれば、大好きなパソコンが1万台は買えるでしょう。1万台のパソコンをどこに置くかはともかく、カジノに行ったことのないあなたは、ゲームのルールから勉強しなければいけません。まずは、ブラック・ジャックのルールを勉強しましょう。



ブラック・ジャックは 1～13 までの数が書かれたカードを使ってゲームをします。各カードの点数は次のように決まっています。

- 1は1点あるいは11点
- 2から9までは、書かれている数の通りの点数
- 10から13までは、10点

このゲームには親を含めた何人かの参加者がおり、それぞれが何枚かのカードの組を持っています。このカードの組のことを手と呼びます。手の点数はカードの点数の合計です。その計算は次のように行うものとします。

- カードの点数の合計が21より大きくなる時は、手の点数を0点とする
- カードの点数として、1は1点と計算しても11点と計算してもよいが、手の点数が最大となる方を選ぶこととする

配られたカードの情報を入力とし、手の点数を出力するプログラムを作成してください。

ただし、1つの手に含まれるカードの枚数 n は1以上21以下とし、同じ数が書かれたカードを何枚でも含むことができるものとします。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 第1のカード 第2のカード … 第 n のカード（整数 整数 …；半角空白区切り）

出力

入力データセットごとに手の点数（整数）を出力します。

入力例	出力例
1	11
7 7 7	21
7 7 8	0
12 1	21
10 1 1	12
0	

問題13 お弁当 (50点)

お昼に食べるお弁当を作るために、お店で食べ物を買いました。お店では、食べ物を入れるのに細長い袋しかもらえなかったので、すべての食べ物を縦に積んで袋に入れる必要があります。袋が倒れにくいように、できるだけ重い物を下にして詰めたいのですが、食べ物の中にはやわらかい物もあって、上に重い物を乗せるとつぶれてしまいます。そこで、それぞれの食べ物がつぶれないように、かつ出来るだけ重い物が下になるように食べ物を積みたいと思います。そこで、食べ物の情報を入力とし、全ての食べ物がつぶれず、かつ全体の重心が最も低くなるような積み方を出力するプログラムを作成してください。



それぞれの食べ物ごとに、名前 f 、自分の重さ w 、上に載せて耐えられる重さ s が指定されます。

「全ての食べ物がつぶれない」というのは、下から順に、 (f_1, f_2, \dots, f_n) と n 個の食べ物を積んだ時、すべての f について、

$$s_{f_i} \geq w_{f_{i+1}} + w_{f_{i+2}} + \dots + w_{f_n}$$

であることを意味します。また、全体の重心 G は、

$$G = (1 \times w_{f_1} + 2 \times w_{f_2} + \dots + n \times w_{f_n}) / (w_{f_1} + w_{f_2} + \dots + w_{f_n})$$

とします。ただし、食べ物の個数 n は1以上10以下とし、食べ物の名前 f は20文字以内の半角英文字列とします。また、解はちょうど1つだけ存在するものとします。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 食べ物の個数 n (整数)
2行目 第1の食べ物の情報 $f w s$ (半角英文字列 整数 整数; 半角空白区切り)
3行目 第2の食べ物の情報
:
:
 $n+1$ 行目 第 n の食べ物の情報

出力

入力データセットごとに食べ物の名前を下から積み上げる順に出力します。

1行目 1番下の食べ物の名前 (半角英文字列)
2行目 下から2番目の食べ物の名前
:
:
 n 行目 1番上の食べ物の名前

入力例	出力例
4 sandwich 80 120 apple 50 200 cheese 20 40 cake 100 100	apple cake sandwich cheese kanzume
9 onigiri 80 300 onigiri 80 300 anpan 70 280 mikan 50 80 kanzume 100 500 chocolate 50 350 cookie 30 80 purin 40 400 cracker 40 160	purin chocolate onigiri onigiri anpan mikan cracker cookie
0	

問題14 サイコロパズル (80点)

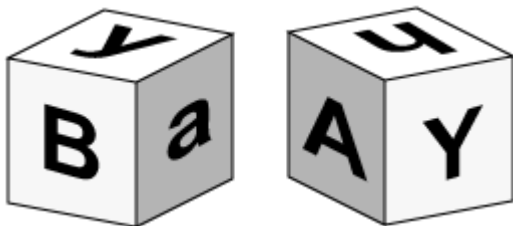
各面にアルファベット一文字(a ~ z、A ~ Z)が描かれたサイコロがあります。



このようなサイコロを8つ組み合わせて $2 \times 2 \times 2$ の立方体を作することを考えます。

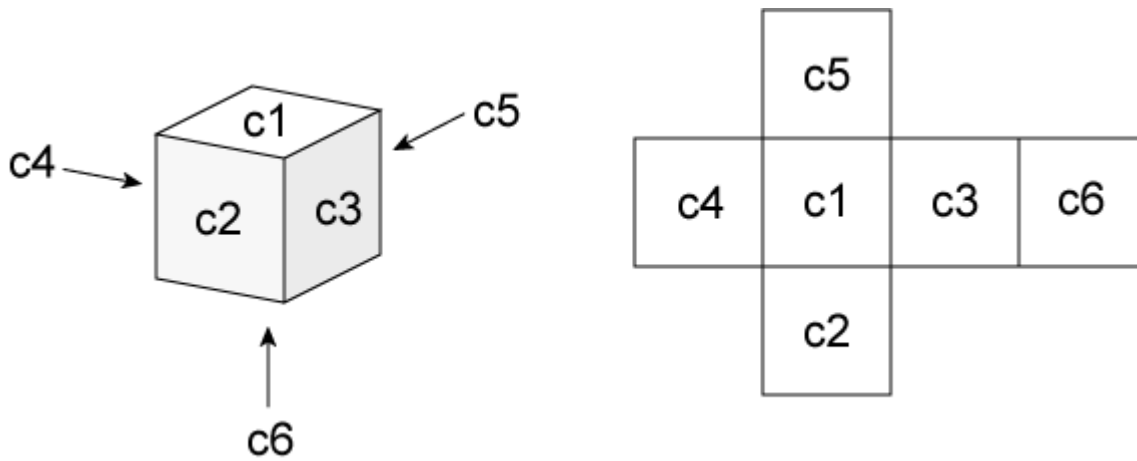


組み合わせ方には条件があり、各サイコロの向き合う面は同じアルファベットでかつ一方が小文字、もう一方が大文字でなければなりません。例えば、a と描かれた面に接することができるのは A と描かれた面です。ただし、接するときの文字の向きは問いません。



このルールに従い、8つのサイコロの情報を入力とし、立方体を作れるか否かを判定し出力するプログラムを作成してください。立方体を作れる場合は YES (半角英大文字)、作れない場合は NO (半角英大文字) と出力してください。

なお、サイコロの各面の文字を次の図にあるように c1~c6 と表すことにします。また、1つのサイコロに同じ文字が複数回描かれていることは無いものとします (同じアルファベットの大文字と小文字はその限りではありません)。



プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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 1つ目のサイコロ情報 s (半角英文字列)

s は長さ6の文字列でありサイコロの各面との対応は以下の通りです

1 文字目 : $c1$

2 文字目 : $c2$

:

6 文字目 : $c6$

2行目 2つ目のサイコロ情報

:

:

8行目 8つ目のサイコロ情報

出力

入力データセットごとに判定結果 (半角英大文字) を出力します。

入力例	出力例
zabznq	YES
BCxmAi	NO
ZcbBCj	YES
aizXCm	YES
QgmABC	NO
JHzMop	
ImoXGz	
MZTOhp	
zabznQ	
BCxmAi	
ZcbBCj	
aizXCm	
QgmABC	
JHzMop	
ImoXGz	
MZTOhp	
abcdef	
ABDCFE	
FBDCAE	
abcdef	
BEACDF	
bfcaed	
fabcde	
DEABCF	
UnivOf	
AizuaH	
zTXZYW	
piGIIt	
GRULNP	
hiGgth	
uAzIXZ	
FiZmKZ	
UnivOf	
AizuaH	
piGIIt	
hiGgth	
GRULNP	
uAzIXZ	
FiZmKZ	
ZTXZYW	
0	

問題15 博士の研究室 (80点)

会津大学の鶴賀博士はとても研究熱心なことで有名です。彼の研究室には複数の学生がいますが、彼は毎日夜遅くまで研究をするので必ず最後に帰宅します。彼の研究室にはドアでつながった複数の部屋があり、最後に研究室を出る人がすべての部屋の明かりを消して帰宅することになっています。最近、大学では省エネに力を入れているため、すべての明かりが消えているか厳しくチェックされます。困ったことに彼は誰もが認める極度の臆病者で、明かりの消えた部屋には決して入ることができません。そこで、彼はある部屋の照明のON/OFFを他の部屋から操作できるように研究室の照明設備を改造しました。ところが、予算の都合で1つの部屋からコントロールできる部屋が限られています。その上、帰宅時の各部屋の明かりの状態も日によって異なるため、全ての明かりを消して出口までたどりつくのに毎日かなりの時間が掛かっています。そこで、研究室で一番年下のあなたが博士を助けるためにプログラムを作成することになりました。

研究室の部屋情報、各部屋の明かりの点灯情報、各部屋の照明スイッチの情報を入力とし、博士がすべての明かりを消して帰宅できるかどうかを出力するプログラムを作成してください。ただし、部屋の数 n は1以上15以下の整数、ドアの数 m は1以上30以下の整数、照明の点灯情報 i は 0 または 1 の整数でそれぞれ消灯と点灯を表し、各部屋には1以上 n 以下の整数で番号が付与されているものとします。出口は番号 n の部屋に存在し、博士は常に部屋 1 からスタートするものとします。

なお、出力は、博士の取るべき行動に応じて以下の3通りに分けて出力します。

- ケース1. 出口以外の全ての明かりを消して出口にたどりつける場合（経路の過程で出口を通ってもよい）。

You can go home in X steps.

と出力します。ここで X は部屋の移動、スイッチのON/OFF をそれぞれ1ステップとした場合の、博士の部屋から出口にたどり着くまでの最短のステップ数です。さらに、以下の文字列に従い博士の部屋から出口までの経路（博士の取るべき行動）を X 行で出力します。

- 部屋 R へ移動する場合

Move to room R .

- 部屋 R の照明を消す場合

Switch off room R .

- 部屋 R の照明を点灯する場合

Switch on room R .

ここで、 R は部屋の番号を表します。博士がある部屋に移動した直後、複数のスイッチを操作して次の部屋に移動する場合は操作する部屋番号が小さいほうから出力してください。この条件を満たす限り、複数の解がある場合は、どれを出力してもかまいません。

この情報をもとに、博士は無事帰宅することができます。

- ケース2. 出口にたどり着くことはできるが、出口のある部屋以外の全ての明かりを消すことができない場合。

You can not switch off all lights.

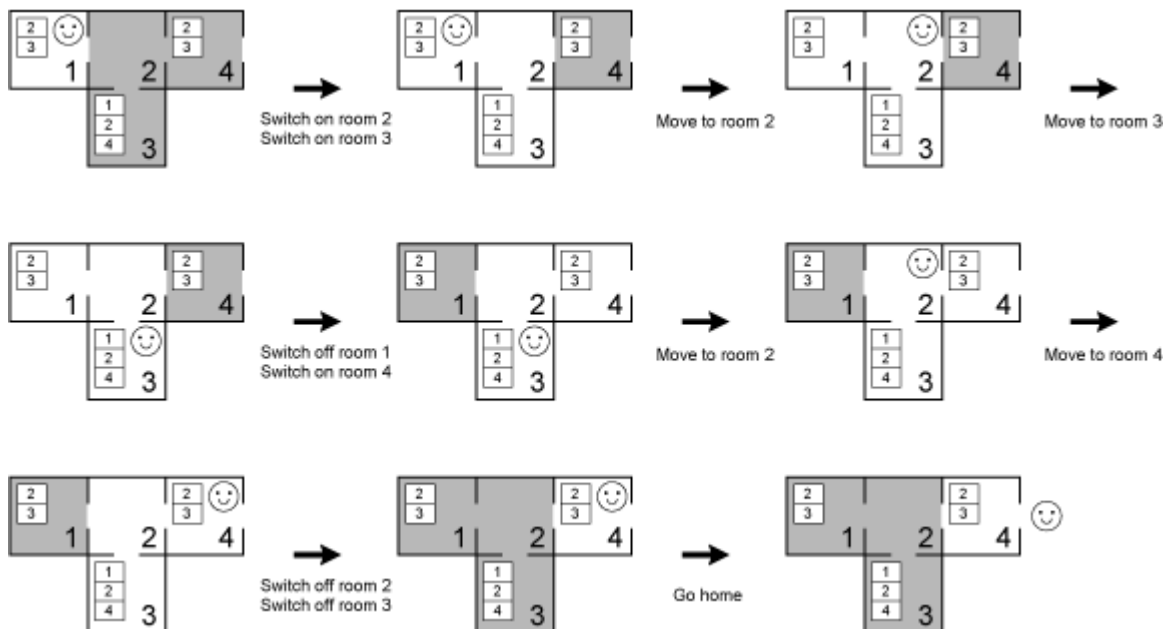
と出力する。この場合、博士は省エネを守らなかったとして大学に罰せられます。

- ケース3. どうあがいても出口にたどり着けない場合。

Help me!

と出力する。この場合、博士は警備員に緊急救助を求めます。

簡単な例を示します。この例では、研究室は4つの部屋で構成され、部屋1と2、2と3、2と4がそれぞれ繋がっています。また、部屋1及び4から部屋2、3の照明を操作することができ、部屋3から部屋1、2、4の照明を操作することができます。最初、部屋2、3、4の照明が消えた状態で、博士が部屋1にいます。



この状況では、博士が取るべき行動は次のようになります。

- 部屋2と3の照明をonにする。
- 部屋2に移動する。
- 部屋3に移動する。
- 部屋1の照明をoffにし、部屋4の照明をonにする。
- 部屋2に移動する。
- 部屋4に移動する。
- 部屋2と3の照明をoffにする。

これで博士は部屋4以外の照明を消すことができ、帰宅することができます。

プログラムは以下に定義する入力が続く限り処理を繰り返し、入力が終わったら終了するように作成してください。

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2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 部屋の数 n ドアの数 m (整数 整数 ; 半角空白区切り)
2行目 1個目のドアの情報 $s t$ (整数 整数 ; 半角空白区切り)
各記号の意味は以下の通りです。
 $s t$: 部屋 s と部屋 t がドアで繋がっている
3行目 2個目のドアの情報
4行目 3個目のドアの情報
:
:
m+1行目 m個目のドアの情報
m+2行目 照明の点灯情報 $i_1 i_2 \dots i_n$ (整数 整数 \dots ; 半角空白区切り)
各記号の意味は以下の通りです。
 i_1 : 部屋の1の照明の情報
 i_2 : 部屋の2の照明の情報
:
 i_n : 部屋のnの照明の情報
m+3行目 1つ目の部屋のスイッチ情報 $k r_1 r_2 \dots r_k$ (整数 整数 \dots ; 半角空白区切り)
各記号の意味は以下の通りです。
 k : スイッチの数
 r_1, r_2, \dots, r_k : 照明を操作できる部屋番号
m+4行目 2つ目の部屋のスイッチ情報
:
:
m+k+2行目 k個目の部屋のスイッチ情報

出力

入力データセットごとに、上記の3通りの結果に応じて以下の形式で出力します。

- ケース1の場合
1行目 You can go home in X steps. (半角英文字列)
2行目 1つ目の博士が取るべき行動 (半角英文字列)
3行目 2つ目の博士が取るべき行動
:
X+1行目 X個目の博士が取るべき行動
- ケース2の場合
1行目 You can not switch off all lights. (半角英文字列)
- ケース3の場合
1行目 Help me! (半角英文字列)

入力例	出力例
4 3	You can go home in 10 steps.
1 2	Switch on room 2.
2 3	Switch on room 3.
2 4	Move to room 2.
1 0 0 0	Move to room 3.
2 2 3	Switch off room 1.
0	Switch on room 4.
3 1 2 4	Move to room 2.
2 2 3	Move to room 4.
4 3	Switch off room 2.
1 2	Switch off room 3.
2 3	You can not switch off all lights.
2 4	Help me!
1 0 0 0	
2 2 3	
0	
3 1 2 4	
1 3	
4 3	
1 2	
2 3	
2 4	
1 0 0 0	
2 2 3	
0	
2 1 2	
2 2 3	
0 0	