1 Boxer Billy
Input File: BoxerBillyIn.txt

After paying his college tuition Billy is short on funds. Therefore, he wants to purchase the minimum number of boxes, all of the same capacity, to put his books in. All of his books are the same size. Billy's frugal boxing style is to completely fill up a box, before packing another box.

Given the number of books that will fit into a box, and the number of books Billy wants take with him, determine how many boxes Billy should purchase, and how many books will be in the last box he packs.

Inputs:
There will be one line of input that contains two integers separated by a space. The first integer represents the maximum number of books that fit into a box, and the second integer represents the number of books Billy wants to box.

Output:
There will be one line of output that gives the minimum number of boxes Billy should purchase, and the number of books that will be in last box he packs. The output should be annotated exactly as shown below.

Sample Input
```
7 275
```

Resultant Output
Purchase 39 boxes and put 2 books in the last box packed

2 Save Our Planet
Input File: SaveOurPlanetIn.txt

Inspired by SavingOurPlanet.net, Maggie has replaced her gasoline engine car with a hybrid vehicle in order to both reduce her carbon footprint and save a lot of money. The hybrid vehicle is powered by either its electric motor, whose electric is supplied by the car's rechargeable battery, or its gasoline engine. Every day of the year (356 days, on Maggie's planet there are no leap years) she travels the same number of miles, and then fully charges the car's battery while she is sleeping.

The new car's electric motor powers the vehicle for the first 25 miles, at which point the battery that supplies electric to it is fully discharged. Then the car is powered by an efficient gasoline engine whose EPA fuel consumption rating is 55 miles per gallon. Considering the cost of gasoline and the cost of electricity to charge the car's battery, during the first 25 miles when the car is being powered by the electric motor the EPA rates the car's equivalent fuel consumption at 135 miles per gallon.

Given the number of miles Maggie drives each day of the year, the cost per gallon of gasoline, and the EPA miles per gallon rating of the car she replaced, your task is to compute the equivalent miles per gallon of her hybrid vehicle, the reduction in her annual fuel cost, and the total reduction in fuel cost over the 200,000 mile life of the car.

Inputs:
The first line of input will be the number cases to consider. This will be followed by one line of input per case containing three real numbers. The first number will be the number of miles Maggie drives each day of the year, the second number will be the price of gasoline (dollars per gallon), and the third number will be the EPA's rated miles per gallon of the car she replaced. All inputs on a line will be separated by a space

Outputs:
There will be one line of output per case that contains three real numbers with two digits of precision rounded up. The first number will represent the equivalent miles per gallon of the hybrid vehicle. The second number will represent the annual gasoline cost savings in dollars. The third number will represent the total gasoline cost savings over the 200,000 mile life of the hybrid vehicle. All calculations will be based on the EPA miles per gallon ratings, and the outputs on a line will be separated by a space.

Sample Input

```
6
25.0 2.45 28.8
25.0 2.45 20.0
30.0 2.45 28.8
30.0 3.45 28.8
40.0 3.45 28.8
40.0 3.45 20.0
```

Resultant output:

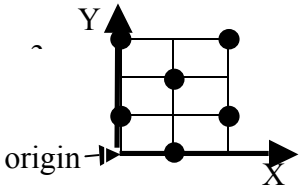
```
135.00 610.66 13384.26
135.00 952.21 20870.37
108.66 684.61 12504.35
108.66 964.05 17608.16
87.35 1172.33 16059.34
87.35 1941.87 26601.01
```

3 Field Trips
Input File: FieldTripsIn.txt

Breanne, the bunny, is making trips to destination points in a rectangular field that is **w** units wide and **d** units deep. The field has a Cartesian coordinate system cut into it. On her trips, she can only move according to the following rules:

1. She must remain on the field during her entire trip
2. She always starts her trips at the origin, and each trip consists of **m** movements
3. Each of the **m** movements is always 1 unit in either the positive **x**-axis direction, or the positive **y**-axis direction
4. On any given trip she must make 2**n** - 1 = **m** movements, with **n** an integer >= 1

The dots in the below figure represent the six possible destination points Breanne could arrive at in a 2 x 3 field, with **n** limited to the values 1, 2, and 3 to keep her on the field.



Your task is to determine the total number of different (x, y) destinations Breanne could arrive at on a field of given width, **w**, and given depth, **d**.

Inputs:
The first line of input will be the number of fields to consider. This will be followed by one line per field that contains two integers separated by a space that represent the width, **w**, and depth, **d**, of the field.

Outputs:
There will be one line of output per field that contains the total number of different destination (x, y) points Breanne could arrive at.

Sample Inputs:
```
3
2 3
6 8
17 10
```

Resultant Output:
```
6
31
99
```

4 Look and Say
Input File: LookAndSayIn.txt

Ryan likes to recite numeric Look-and-Say sequences. The first element of these sequences, E_1, is a given non-negative integer seed value. Any other element of the sequence, E_n, is generated from its predecessor, E_{n-1}, as follows: Starting on the left side of element E_{n-1}, group adjacent occurrences of the same digit and then, for each grouping, use one digit to indicate how many digits are in the grouping followed by which digit the group contains.

For example, given a seed value of 1, the first eight elements of the sequence, E_1 through E_8, are:

1, 11, 21, 1211, 111221, 312211, 13112221, 1113213211

because when Ryan is given a seed value of 1, he looks at it and says "one 1", which becomes the second element of the sequence: 11. Then he looks at that element and says "two 1's", which becomes the third element: 21. Looking at the third element he says "one 2" then "one 1", which becomes the fourth element: 1211. Looking at the fourth element of the sequence he says "one 1" then "one 2" then "two 1's", which becomes the fifth element of the sequence: 111221. He continues in this way until he has recited a given number of terms, **`t`**. In the above sequence **`t`** is 8.

Your task is to generate the first **`t`** terms of the Look-and-Say sequence given its non-negative integer seed value, E_1, and the number of terms to generate, **`t`**.

Inputs:
The first line contains a positive integer indicating how many sequences are to be generated. This is followed by one line per sequence that contains two positive integers separated by a space. The first integer is the sequence's seed value, and the second integer is the number of terms, **`t`**, of the Look-and-Say sequence to generate.

Outputs:
For each Look-and-Say sequence described in the input, the program should generate a single line containing the first **`t`** elements of that sequence, each separated by a space.

Sample input:
```
5
1 8
22 6
55555333337 5
0 4
34 1
```

Resultant output:
1 11 21 1211 111221 312211 13112221 1113213211
22 22 22 22 22 22
55555333337 555317 35131117 131511133117 1113111531232117
0 10 1110 3110
34

5 On Vacation
Input File: OnVacationIn.txt

Entrepreneur Ada always divides the 365 days of a year into **n** profit periods each consisting of consecutive days. The first of these profit periods, P_1 always begins on January 1st, and profit period P_{i+1} always begins the day after profit period P_i ends. Each year her accountant gives her a prediction of the profit for each of the **n** profit periods, which for some periods could be negative (a net loss).

In order to attract the best employees, each year Ada places her employees on vacation for the maximum number of pay periods that does not include the most profitable consecutive profit periods. This could mean that the store could be open during all profit periods. Your task is to determine the fewest number of consecutive profit periods, during which the business would be open, that produces the maximum profit for the year.

For example, given the below 17 net profits for periods P_1 through P_{17} as:

0 13 -3 -25 20 -3 -16 -23 18 20 -7 12 -5 -22 15 -4 7

the first and last profit periods the business would open would be P_9 and P_{12} respectively, which yields a maximum net profit for the year of 43 (= 18 + 20 + (-7) + 12).

Inputs:
The first line of input will be the number of years to consider. This will be followed by two lines of input per year. The first line contains one integer that represents the number of profit periods for that year, **n**. The second line contains **n** integers that represent predicted profit for each profit period in ascending profit period number order (P_1, P_2, P_3, …). Multiple inputs on a line are separated by a space.

Outputs:
There will be one line of output per year that contains three integers each separate by a space. The first integer represents the maximum attainable profit for the year over a group of consecutive profit periods. The second and third integers represent the first and last profit period numbers that produce that profit within a minimum number of consecutive profit periods.

Sample Input	**Resultant Output**
4 17 0 13 -3 -25 20 -3 -16 -23 18 20 -7 12 -5 -22 15 -4 7 17 20 -3 -25 20 -12 15 -23 12 20 -7 14 -5 -22 15 -4 6 10 12 5 3 -2 12 -22 18 -5 -11 15 -3 -1 2 12 5 3 -2 12 -14 18 -5 -11 15 -6 6 2	43 9 12 39 8 11 18 6 6 23 1 12

6 Narrow Gallery
Input File: NarrowGalleryIn.txt

An art gallery is laid out as **N** rows of two rooms side-by-side, as shown below for **N** = 10. Doors connect all adjacent rooms (north-south and east-west, but not diagonally). The curator, Skyler, has been told that she must close off **K** of the rooms because of staffing cuts. However, visitors must be able to enter at least one of the two rooms at one end of the gallery, proceed through the gallery, and exit from at least one of the two rooms at the other end. Therefore, Skyler cannot close off any two rooms that would block passage through the gallery. That is, she may not block off two rooms in the same row or two rooms in adjacent rows that touch diagonally.

Furthermore, Skyler has determined how much value each room has to the general public (the integers in the gallery shown below), so that she can close off those **K** rooms that leave the most total value available to the public, without blocking passage through the gallery (shaded rooms shown below for **K** = 5).

7	8
4	9
3	7
5	9
7	2
10	3
0	10
3	2
6	3
7	9

Given the number of rows in a gallery, **N**, the number of rooms to close off, **K**, and the value of each room, your task is to determine which **K** rooms Skyler should close that leave the most total value available to the public.

Inputs:
The first line of input will be the number of galleries to consider. This will be followed by two sets of inputs per gallery. The first set contains two integers on one line that represent the number of rows in the gallery, **N**, followed by the number of rooms to close, **K**. The second set of inputs is **N** lines containing two integers each that represent the values of two adjacent rooms in the gallery. Multiple inputs on a line are separated by a space.

Outputs:
There will be one integer output per gallery that represents the gallery's total value to the public after the **K** rooms, that maximizes this value and still permits visitors to enter the gallery from one end and leave from the other end, has been closed.

(Sample inputs and outputs are shown on the next page)

Sample Input:

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

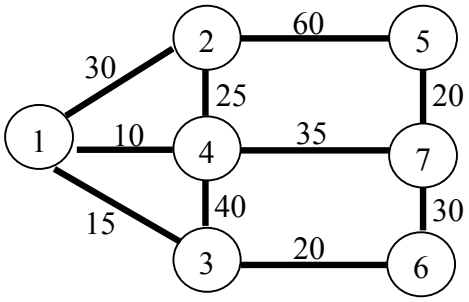
Resultant Output:

```
102
12
18
```

7 Weak Bridges
Input File: WeakBridgesIn.txt

A group of tourists are to be transported by bus from one city to another within a country. The cities in the country are connected by two way bridges of different strengths such that each bridge can support a bus carrying a given maximum number of people (the passengers *plus* the bus driver). The below figure shows seven cities (the circles) connected by 10 bridges (the lines) whose *people* capacities are written adjacent to the lines.

A bridge will collapse if a bus is driven on it carrying more people than it can support. For example, the bridge from city 2 to city 5, in the below figure, cannot support a bus carrying 60 tourists *and* the bus driver.



The people capacity of the bus, tourists plus the driver, is always equal to the largest bridge capacity. Your task is to determine the minimum number of trips required to safely transport a given number of tourists from their home city, to their destination city. You may assume that there will always be at least one route from their home to their destination, and that the cities are numbered consecutively beginning with 1 (i.e., 1, 2, …). If multiple trips are required, all trips except the last trip will transport the same number of tourists.

Inputs
The first line of input will be the number of countries to consider. This will be followed by two sets of inputs per country. The first input set contains three integers on one line that represent the tourists' home city number, followed by their destination city number, followed by the total number of tourists to be transported. The first line of the second set of inputs contains two integers on one line that represent the number of cities in the country, followed by the number of bridges that connect the cities, **`B`**. This line is followed by **`B`** lines (one line per bridge) each of which contains three integers. The first two integers represent the city numbers the bridge connects, and the third integer is the maximum *people* capacity of the bridge. Multiple numbers on a line are separated by a space.

Outputs
There will be one line of output per country containing two integers separated by a space. The first integer represents the minimum number of trips required to safely transport the all of the tourists from their home town to their destination town, and the second integer represents the number of tourists on the bus during the last trip.

(Sample inputs and outputs are shown on the next page)

Sample Input

```
3
7 1 97
7 10
1 2 30
1 3 15
1 4 10
2 4 25
2 5 60
3 4 40
3 6 20
4 7 35
5 7 20
6 7 30
5 6 84
7 9
4 7 30
3 4 15
1 4 22
3 5 20
1 7 10
5 7 28
1 3 40
3 6 50
5 2 30
1 2 105
6 8
1 2 11
1 6 12
6 4 10
2 3 17
2 6 9
3 4 37
4 5 13
5 6 19
```

Resultant Output

```
5 1
4 21
10 6
```