HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN · FH
MÜNCHEN

Kombinatorik

Binomialkoeffizient

Pascalsches Dreieck

Catalan-Zahlen

Gliederung

- Was ist Kombinatorik
- Die Fakultät + Beispiel 10338
- Binomialkoeffizient + Beispiel 10213
- Pascalsches Dreieck + Beispiel 485
- Catalan Zahlen + Beispiel 10007
- Beispiel 11270

Was ist Kombinatorik?

- Beschäftigt sich mit endlichen oder abzählbar unendlichen Diskreten Strukturen
- Der bekannteste Bereich ist die abzählende Kombinatorik
- Abzählende Kombinatorik beschäftigt sich mit der Bestimmung der Anzahl möglicher Anordnungen oder Auswahlen

Die Fakultät

- Wird durch ein dem Argument nachgestelltem Ausrufezeichen („!“) abgekürzt
- Ist das Produkt aller natürlichen Zahlen kleiner oder gleich der gewählten Zahl

$$n! = 1 * 2 * 3 * \ldots * n = \prod_{k=1}^{n} k$$

Beispiel 10338 Mischievous Children

Problem C

Mischievous Children

Input: standard input
Output: standard output
Time Limit: 1 second
Memory Limit: 32 MB

Adam's parents put up a sign that says "CONGRATULATIONS". The sign is so big that exactly one letter fits on each panel. Some of Adam's younger cousins got bored during the reception and decided to rearrange the panels. How many unique ways can the panels be arranged (counting the original arrangement)?

Input / Output

The first line of input is a single non-negative integer. It indicates the number of data sets to follow. Its value will be less than 30001.

Each data set consists of a single word, in all capital letters. For each word, output the number of unique ways that the letters can be rearranged (counting the original arrangement). Use the format shown in Sample Output, below.

Each word will have at most 20 letters. There will be no spaces or other punctuation.

The number of arrangements will always be able to fit into an `unsigned long int`. Note that 12! is the largest factorial that can fit into an `unsigned long int`.

Sample

Sample Input

```
3
HAPPY
WEDDING
ADAM
```

Sample Output

```
Data set 1: 60
Data set 2: 2520
Data set 3: 12
```

Zu beachten:

- Maximal 20 Buchstaben pro Wort
  - 20! = 2.432.902.008.176.640.000 → Long verwenden!
- Bei mehrmaligem vorkommen eines Buchstabens teilen

Binomialkoeffizient

- Beschreibt die Anzahl der Möglichkeiten k Elemente aus n zu wählen:

$$\binom{n}{k} = \frac{n!}{k! * (n-k)!}$$

- Beispiel: „Lotto – 6 aus 49“

$$\binom{49}{6} = \frac{49!}{6! * (49-6)!} = \frac{49*48*47*46*45*44}{6*5*4*3*2*1} =$$

13983816 Möglichkeiten

10213 - How Many Pieces of Land?

Problem G

How Many Pieces of Land?

Input: Standard Input
Output: Standard Output
Time Limit: 3 seconds

You are given an elliptical shaped land and you are asked to choose **n** arbitrary points on its boundary. Then you connect all these points with one another with straight lines (that's **n*(n-1)/2** connections for **n** points). What is the maximum number of pieces of land you will get by choosing the points on the boundary carefully?

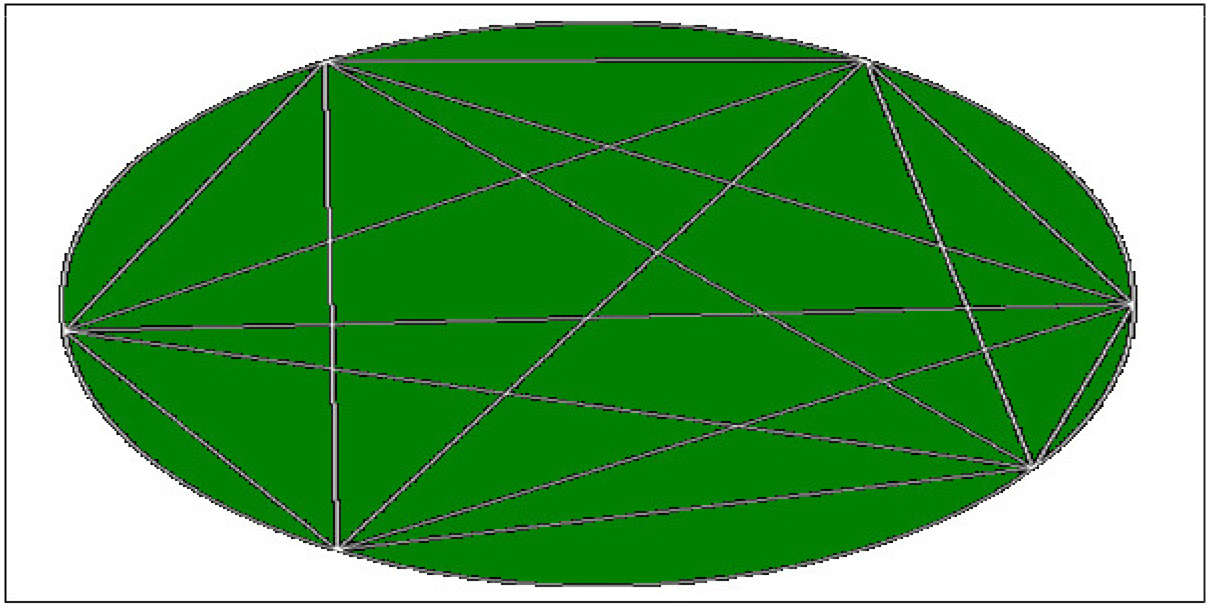


Fig: When the value of n is 6.

Input

The first line of the input file contains one integer **S (0 < S < 3500)**, which indicates how many sets of input are there. The next **S** lines contain **S** sets of input. Each input contains one integer **N (0<=N<2^31)**.

Output

For each set of input you should output in a single line the maximum number pieces of land possible to get for the value of **N**.

Sample Input:

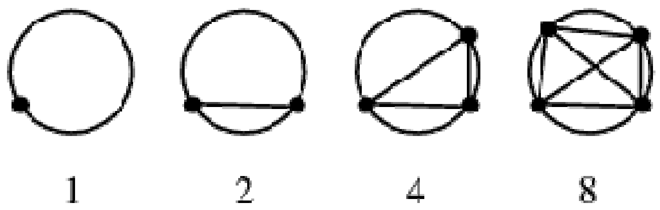
```
4
1
2
3
4
```

Sample Output:

```
1
2
4
8
```

- Entspricht „Mosers Kreis Problem“:

$$g(n) = \binom{n}{4} + \binom{n}{2} + 1$$



Quelle: Wolfram MathWorld

Pascalsches Dreieck

- Ist eine geometrische Darstellung der Binomialkoeffizienten
- Der Name geht auf Blaise Pascal zurück
- Früheste detaillierte Darstellung erschien im 10. Jahrhundert
- Bereits von vor Christus gibt es Fragmente die das Dreieck beschreiben

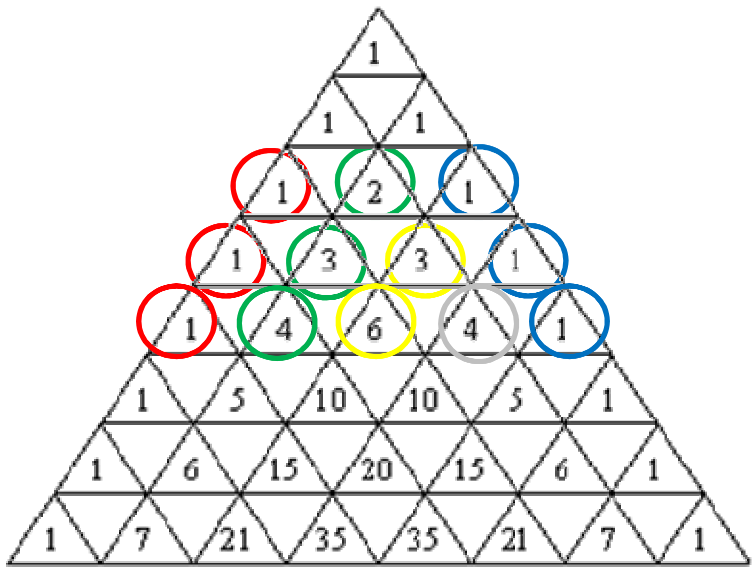
$$
\begin{array}{ccccccccccccccc}
 & & & & & & & 1 & & & & & & & \\
 & & & & & & 1 & & 1 & & & & & & \\
 & & & & & 1 & & 2 & & 1 & & & & & \\
 & & & & 1 & & 3 & & 3 & & 1 & & & & \\
 & & & 1 & & 4 & & 6 & & 4 & & 1 & & & \\
 & & 1 & & 5 & & 10 & & 10 & & 5 & & 1 & & \\
 & 1 & & 6 & & 15 & & 20 & & 15 & & 6 & & 1 & \\
1 & & 7 & & 21 & & 35 & & 35 & & 21 & & 7 & & 1
\end{array}
$$

- errechnet sich durch die Aufaddierung der zwei darüberstehenden Zahlen
- oder über den Binomialkoeffizienten

$$
\begin{array}{ccccccccc}
 & & & & \binom{0}{0} & & & & \\
 & & & \binom{1}{0} & & \binom{1}{1} & & & \\
 & & \binom{2}{0} & & \binom{2}{1} & & \binom{2}{2} & & \\
 & \binom{3}{0} & & \binom{3}{1} & & \binom{3}{2} & & \binom{3}{3} & \\
\binom{4}{0} & & \binom{4}{1} & & \binom{4}{2} & & \binom{4}{3} & & \binom{4}{4}
\end{array}
$$

Vorteil Binomialkoeffizient: beliebige Elemente im Dreieck können ohne Abhängigkeit ausgerechnet werden

•Vorteil des Dreiecks: Binomischer Lehrsatz



Quelle: Uni Linz

$$(a+b)^2 = 1a^2 + 2ab + 1b^2$$

$$(a+b)^3 = 1a^3 + 3a^2b^1 + 3a^1b^2 + 1b^3$$

$$(a-b)^4 = 1a^4 - 4a^3b^1 + 6a^2b^2 - 4a^1b^3 + 1b^4$$

Beispiel 485 Pascal Triangle of Death

Pascal's Triangle of Death

In this problem, you are asked to generate Pascal's Triangle. Pascal's Triangle is useful in many areas from probability to polynomials to programming contests. It is a triangle of integers with ``1'' on top and down the sides. Any number in the interior equals the sum of the two numbers above it. For example, here are the first 5 rows of the triangle.

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

In ``Pascal's Triangle of Death,'' you are to generate a **left justified** Pascal's Triangle. When any number in the triangle is exceeds or equals 10^{60}, your program should finish printing the current row and exit. The output should have each row of the triangle on a separate line with one space between each element.

The final element of each line should be directly followed by a newline. There is no space after the last number on each line.

Sample Input

There is no input for this problem.

Sample Output

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
.
.
.
```

etc.

Zu beachten:

- bis 10^60 → BigInteger verwenden
- beim Erreichen des Wertes muss die letzte Zeile noch fertig geschrieben werden
- am Ende der Zeile KEIN Leerzeichen mehr

Catalan-Zahlen

- Benannt nach Eugène Charles Catalan
- Anwendungen:
  - Zerlegung eines konvexen n(+2)-Ecks in Dreiecke
  - Möglichkeiten öffnender und schließender Klammern
  - Zur Erzeugung vollständiger Binärbäume mit n Knoten (Beispiel 10007)

$$C_n = \frac{1}{n+1}\binom{2n}{n}$$

n	C(n)
0	1
1	1
2	2
3	5
4	14
5	42
6	132
7	429
8	1.430
9	4.862
10	16.796
…	…
20	6.564.120.420
21	24.466.267.020
22	91.482.563.640
23	343.059.613.650
24	1.289.904.147.324

Beispiel 10007 Count the Trees

Count the Trees

Another common social inability is known as ACM (Abnormally Compulsive Meditation). This psychological disorder is somewhat common among programmers. It can be described as the temporary (although frequent) loss of the faculty of speech when the whole power of the brain is applied to something extremely interesting or challenging.

Juan is a very gifted programmer, and has a severe case of ACM (he even participated in an ACM world championship a few months ago). Lately, his loved ones are worried about him, because he has found a new exciting problem to exercise his intellectual powers, and he has been speechless for several weeks now. The problem is the determination of the number of different labeled binary trees that can be built using exactly n different elements.

For example, given one element A, just one binary tree can be formed (using A as the root of the tree). With two elements, A and B, four different binary trees can be created, as shown in the figure.

If you are able to provide a solution for this problem, Juan will be able to talk again, and his friends and family will be forever grateful.

Input

The input will consist of several input cases, one per line. Each input case will be specified by the number n ($1 \leq n \leq 300$) of different elements that must be used to form the trees. A number `0` will mark the end of input and is not to be processed.

Output

For each input case print the number of binary trees that can be built using the n elements, followed by a newline character.

Sample Input

```
1
2
10
25
0
```

Sample Output

```
1
4
60949324800
75414671852339208296275849248768000000
```

Anzahl der Möglichkeiten n Zeichen zu verteilen

Catalan-Zahl

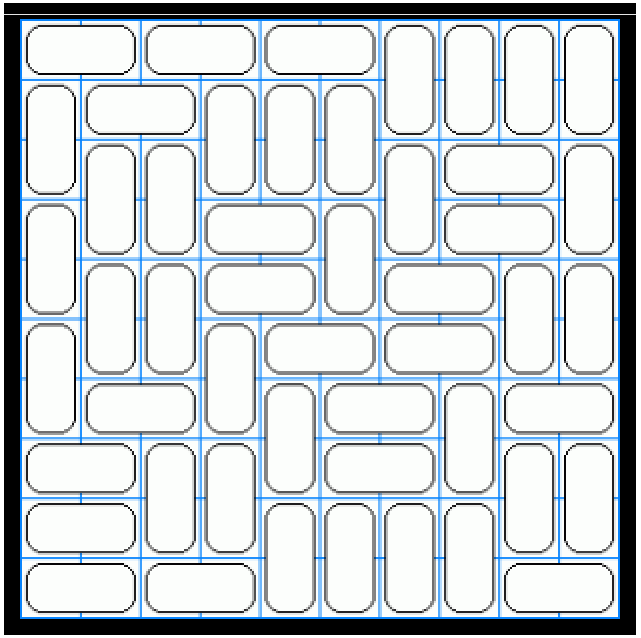
$$Result(n) = n! * \frac{1}{n+1} * \binom{2n}{n} = \frac{(2n)!}{(n+1)!}$$

Beispiel 11270 Tiling Dominoes

A. Tiling Dominoes

Problem

Given a rectangular grid, with dimensions **m x n**, compute the number of ways of completely tiling it with dominoes. Note that if the rotation of one tiling matches another, they still count as different ones. A domino is a shape formed by the union of two unit squares meeting edge-to-edge. Equivalently, it is a matching in the grid graph formed by placing a vertex at the center of each square of the region and connecting two vertices when they correspond to adjacent squares. An example of a tiling is shown below.



The Input

The input will consist of a set of lines with **m n**, given the restriction **n*m<101**.

The Output

For each line of input, output the number of tilings in a separate line.

Sample Input

```
2 10
4 10
8 8
```

Sample Output

```
89
18061
12988816
```

- Temperley & Fisher:

$$\prod_{j=1}^{N}\prod_{k=1}^{N}\left(4\cos^2\frac{\pi j}{2n+1}\,4\cos^2\frac{\pi k}{2n+1}\right)$$

Quellen:

- Logofatu, Doina: Algorithmen und Problemlösungen mit C++
- Wolfram Alpha: http://www.wolframalpha.com/
- http://uva.onlinejudge.org/