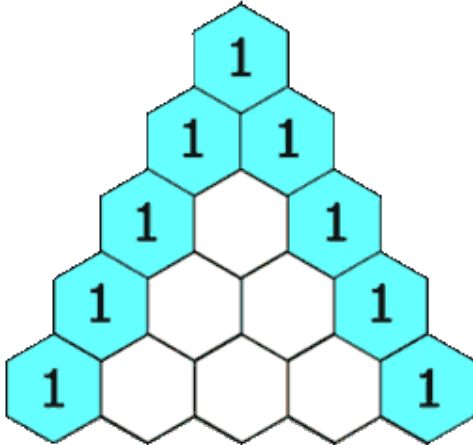



Square routes

You may have heard of **Pascal's triangle** before. A triangular array that is constructed in the following way. At the top of the triangle you write the number 1. Each next row of the triangle is constructed by adding the number above and to the left with the number above and to the right of each cell in the row to find the new value in that cell. If either the number to the right or left is not present, substitute a zero in its place. Below we demonstrate how the first five rows of the triangle are constructed.



However, few people know that it would historically be more correct to talk about **Pascal's square**. Blaise Pascal (1623-1662) — the French mathematician after which the triangle was named — arranged the numbers himself in the form of a square. The construction of the square is analogous to that of the triangle. Pick a number $n \in \mathbb{N}_0$. Fill all positions on the top row and the left column with that number n . Traverse all other cells of the square from left to right and from top to bottom, and fill each cell with the sum of the numbers in the two neighbouring cells to the left and on top. Below we show Pascal's square with four rows and four columns, with $n = 1$.

	1	1	1
1	2	3	4
1	3	6	10
1	4	10	20

A property of Pascal's square (resp. triangle) is that each cell displays the number of distinct paths that lead to that cell from the upper left cell of the square (resp. top cell of the triangle), assuming only rightward and downward (resp. left down and right down) movements are allowed. So, in the square given above, the cell marked 4 in the second row, fourth column, can be reached in four ways: 1-1-1-4, 1-1-3-4, 1-2-3-4 and 1-2-3-4.

Assignment

- Write a function `square` that takes an integer $m \in \mathbb{N}_0$ as its argument. The function also has a second optional parameter that takes an integer $n \in \mathbb{N}_0$ (default value: $n=1$). The function must return a list of lists that represents Pascal's square with m rows and m columns, whose top row and left column are filled with the number n .
- Write a function `routes` that takes an integer $m \in \mathbb{N}_0$ as its argument. The function also has a second optional parameter that takes an integer $n \in \mathbb{N}_0$ (default value: $n=1$). The function must return a string representation of Pascal's square with m rows and m columns, whose top row and left column are filled with the number n . In this string representation, each number must be right justified over $p + 1$ positions, with p being the number of digits in the value at the bottom right of the square.

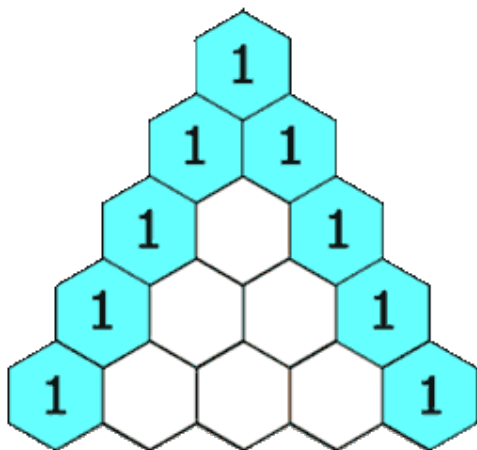
Example

```
>>> square(3)
[[1, 1, 1], [1, 2, 3], [1, 3, 6]]
>>> square(3, 100)
[[100, 100, 100], [100, 200, 300], [100, 300, 600]]
>>> square(4)
[[1, 1, 1, 1], [1, 2, 3, 4], [1, 3, 6, 10], [1, 4, 10, 20]]


>>> print(routes(3))
1 1 1
1 2 3
1 3 6
>>> print(routes(3, 100))
100 100 100
100 200 300
100 300 600
>>> print(routes(4))
1 1 1 1
1 2 3 4
1 3 6 10
1 4 10 20
>>> print(routes(6))
1 1 1 1 1 1
1 2 3 4 5 6
1 3 6 10 15 21
1 4 10 20 35 56
1 5 15 35 70 126
1 6 21 56 126 252
>>> print(routes(8))
1 1 1 1 1 1 1 1
1 2 3 4 5 6 7 8
1 3 6 10 15 21 28 36
1 4 10 20 35 56 84 120
1 5 15 35 70 126 210 330
1 6 21 56 126 252 462 792
1 7 28 84 210 462 924 1716
1 8 36 120 330 792 1716 3432
>>> print(routes(10))
1 1 1 1 1 1 1 1 1 1
1 2 3 4 5 6 7 8 9 10
1 3 6 10 15 21 28 36 45 55
1 4 10 20 35 56 84 120 165 220
1 5 15 35 70 126 210 330 495 715
```

1 6 21 56 126 252 462 792 1287 2002
 1 7 28 84 210 462 924 1716 3003 5005
 1 8 36 120 330 792 1716 3432 6435 11440
 1 9 45 165 495 1287 3003 6435 12870 24310
 1 10 55 220 715 2002 5005 11440 24310 48620

Waarschijnlijk heb je ooit wel al eens gehoord van de **driehoek van Pascal**. Deze driehoek wordt op de volgende manier opgebouwd. Helemaal bovenaan schrijf je het getal 1. De volgende rij van de driehoek wordt telkens gevormd door op elke positie de som van de twee bovenliggende getallen te schrijven. Indien het getal linksboven of rechtsboven ontbreekt, dan vervang je het door een denkbeeldige nul. Hieronder tonen we alvast hoe de eerste vijf rijen van de driehoek opgebouwd worden.



Weinigen weten echter dat we eigenlijk zouden moeten spreken van het **vierkant van Pascal**. Blaise Pascal (1623-1662) — de Franse wiskundige naar wie de driehoek vernoemd is — rangschikte de getallen immers zelf in de vorm van een vierkant. De constructie van het vierkant verloopt analoog aan deze van de driehoek. Kies een getal $n \in \mathbb{N}_0$. Vul alle posities op de eerste rij en de eerste kolom op met het getal n . Doorloop daarna de overige cellen van links naar rechts en van boven naar onder, en vul elke cel met de som van de getallen in de twee naburige cellen links en boven. Hieronder tonen we het vierkant van Pascal met vier rijen en vier kolommen, waarbij $n = 1$.

 1	1	1	1
1	2	3	4
1	3	6	10
1	4	10	20

Een eigenschap van het vierkant (resp. de driehoek) van Pascal is dat elk getal aangeeft hoeveel verschillende paden je kan volgen vanaf de cel linksboven het vierkant (resp. de cel bovenaan de driehoek) naar de cel waarin het getal staat, als je enkel naar rechts en naar beneden (resp. naar linksonder en naar rechtsonder) mag bewegen. In het bovenstaande vierkant kan je bijvoorbeeld de cel met het getal 4 op de tweede rij bereiken op vier mogelijke

manieren: 1-1-1-4, 1-1-3-4, 1-2-3-4 en 1-2-3-4.

Opgave

- Schrijf een functie vierkant waaraan een getal $m \in \mathbb{N}_0$ moet doorgegeven worden. De functie heeft ook nog een tweede optionele parameter waaraan een getal $n \in \mathbb{N}_0$ kan doorgegeven worden (standaardwaarde: $n=1$). De functie moet een lijst van lijsten teruggeven die een vierkant van Pascal voorstelt met m rijen en m kolommen, waarvan de eerste rij en kolom ingevuld worden met het getal n .
- Schrijf een functie paden waaraan een getal $m \in \mathbb{N}_0$ moet doorgegeven worden. De functie heeft ook nog een tweede optionele parameter waaraan een getal $n \in \mathbb{N}_0$ kan doorgegeven worden (standaardwaarde: $n=1$). De functie moet een stringvoorstelling teruggeven van het vierkant van Pascal met m rijen en m kolommen, waarvan de eerste rij en kolom ingevuld worden met het getal n . In deze stringvoorstelling moet elk getal rechts uitgelijnd worden over $p + 1$ posities, waarbij p het aantal cijfers is in het getal rechts onderaan het vierkant.

Voorbeeld

```
>>> vierkant(3)
[[1, 1, 1], [1, 2, 3], [1, 3, 6]]
>>> vierkant(3, 100)
[[100, 100, 100], [100, 200, 300], [100, 300, 600]]
>>> vierkant(4)
[[1, 1, 1, 1], [1, 2, 3, 4], [1, 3, 6, 10], [1, 4, 10, 20]]
```

```
>>> print(paden(3))
1 1 1
1 2 3
1 3 6
>>> print(paden(3, 100))
100 100 100
100 200 300
100 300 600
>>> print(paden(4))
1 1 1 1
1 2 3 4
1 3 6 10
1 4 10 20
>>> print(paden(6))
1 1 1 1 1 1
1 2 3 4 5 6
1 3 6 10 15 21
1 4 10 20 35 56
1 5 15 35 70 126
1 6 21 56 126 252
>>> print(paden(8))
1 1 1 1 1 1 1 1
1 2 3 4 5 6 7 8
1 3 6 10 15 21 28 36
1 4 10 20 35 56 84 120
1 5 15 35 70 126 210 330
1 6 21 56 126 252 462 792
1 7 28 84 210 462 924 1716
1 8 36 120 330 792 1716 3432
```

```
>>> print(paden(10))
```

```
1 1 1 1 1 1 1 1 1 1
1 2 3 4 5 6 7 8 9 10
1 3 6 10 15 21 28 36 45 55
1 4 10 20 35 56 84 120 165 220
1 5 15 35 70 126 210 330 495 715
1 6 21 56 126 252 462 792 1287 2002
1 7 28 84 210 462 924 1716 3003 5005
1 8 36 120 330 792 1716 3432 6435 11440
1 9 45 165 495 1287 3003 6435 12870 24310
1 10 55 220 715 2002 5005 11440 24310 48620
```