

Magma Assignment 02

Combinatorics and Cryptography

Sets, multisets, sequences

A *partition* of n is a sequence of positive integers $\lambda_1 \leq \lambda_2 \leq \dots, \lambda_t$ such that

$$n = \sum_{i=1}^t \lambda_i. \quad (1)$$

Each integer in Eq. (1) is called a *part*. The number of times the part λ_i appears in the sequence $[\lambda_1, \lambda_2, \dots, \lambda_t]$ is called the *multiplicity* of the part λ_i in the partition $[\lambda_1, \lambda_2, \dots, \lambda_t]$. Using the magma function **Partitions**(n) we can obtain the sequence of all the integer partitions of a given input n .

Task

Implement a function called **filteredPartitions** which takes as inputs two integers n and t and returns:

- the sequence of all the partitions of n in which each part has multiplicity at most t (i.e. it is repeated at most t times),
- the number of such partitions.

You are allowed to use **Partitions** as a subroutine.

Points

Submitting a working solution will give you one point.

Example

An example of a working program will produce:

```
> filteredPartitions(6,4);
[
  [ 6 ],
  [ 5, 1 ],
  [ 4, 2 ],
  [ 4, 1, 1 ],
  [ 3, 3 ],
  [ 3, 2, 1 ],
  [ 3, 1, 1, 1 ],
  [ 2, 2, 2 ],
  [ 2, 2, 1, 1 ],
  [ 2, 1, 1, 1, 1 ]
]
> filteredPartitions(6,1);
[
  [ 6 ],
  [ 5, 1 ],
  [ 4, 2 ],
  [ 3, 2, 1 ]
]
```