Lists (1)

You have to program several functions. In each case, few lines of code are enough.

1. Write an integer function `count_diff(f)` that given a list of integers `f` returns the number of different values in the list.

2. Write a float function `product(u, v)` that given two float lists representing two vectors returns the scalar product. You can assume both lists have the same length and are non empty.

3. Write a function `delete_multiples(k, f)` that provided an integer `k` greater than zero and a list of integers returns the list of numbers in `f` that are not multiple of `k`. Numbers in the resulting list must preserve their relative order in `f`.

4. Write a function `erato(n)` that returns the ordered list of prime numbers that are less than natural `n`. Your code has to implement the algorithm known as Sieve of Eratosthenes. This algorithm is one of the most efficient ways to find all of the smaller primes. It is named after Eratosthenes of Cyrene, a Greek mathematician.

5. Write a function `merge(f, g)` that provided two ordered list of integers `f` and `g` returns an ordered list of integers formed by elements of `f` and `g`. 

Warning: do not use any kind of sorting function.

Scoring

Every function counts 20 points.

Sample session

```python
>>> count_diff([3, -1, 0, 3, 2, 0])
4
>>> product([1/3, 0, -1], [3/2, 1/2, 2])
-1.5
>>> delete_multiples(2, [6, 3, -2, -5, 7])
[3, -5, 7]
>>> erato(30)
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29]
>>> merge([1, 2, 5, 9], [-3, 0, 2, 11, 12, 13])
[-3, 0, 1, 2, 2, 5, 9, 11, 12, 13]
```

Problem information

Author: Jorge Castro
Generation: 2016-09-20 09:25:06

http://www.jutge.org