EE016 Homework #1

Deadline: October 10 2022

Q1: Base q to Decimal: Suppose q is an integer obeying 1<q≤10. Write a function that admits a string s as its input. This string is base q representation of an integer. Thus, the characters in this string is from 0 to q-1. Output the corresponding integer.

```
def base_q_to_decimal(s, q):
    return x
```

Q2: Base q to p: Suppose q,p are integers obeying 1<q,p≤10. Write a function that admits a string s as its input. This string is base q representation of an integer. Thus, the characters in this string is from 0 to q-1. Output a string which is base p representation of this integer.

```
def base_q_to_p(s, q, p):
    return x
```

Q3: Sorted merge. Suppose you are given two lists l1 and l2 whose elements are integers that are sorted in non-decreasing order. Write a function that admits these as input and merges them into a new list whose elements are still sorted.

```
def sorted_merge(l1, l2):
    return l
```

Q4: Multiple sorted merge. Suppose you are given a list of lists l_multiple. Each element of this list is a list of integers sorted in non-decreasing order. Write a function that merges elements of l_multiple into a single sorted list of integers.

```
def multiple_sorted_merge(l_multiple):
    return l_sorted
```

Q5: Root finding. Write a function that takes an integer x as its input. This function should output two integers root and pwr, such that 1 < pwr < 6 and root**pwr is equal to the input x. If there are multiple answers, it should output the answer with the largest pwr. If there is no solution, output False.

```
def find_root(x):
    return root, pwr
```

Q6: Least Common Multiple. Write a function that admits an arbitrary number of integers (e.g. via *args). Return the smallest positive integer lcm such that lcm is divisible by all of the inputs to the function.
Remark: The runtime of the ideal code should be proportional to sqrt(max(a1,…an)) where ai are the inputs.
Hint: Use factorization into prime numbers. Write an integer a=p1^q1*p2^q2*…*pk^qk where pk are prime numbers. Consider making this factorization a separate function.

```
def least_common_multiple(*args):
    return lcm
```

Q7: Greatest Common Divisor: Write a function that admits an arbitrary number of integers. Return the largest positive integer gcd such that gcd divides all of the inputs to the function.
Hint: You can still use the factorization in **Q6**.

```
def greatest_common_divisor(*args):
    return gcd
```

Q8: Return unique values: Write a function that admits a list of lists L and outputs its unique values as a list of integers. For instance,

Input: [[1,2,3], [4,6,2], [3,6,4,1]]
Output: [1,2,3,4,6]

```
def unique_values(L):
    return unique_list
```