E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Problem: Given string of length n, consisting of lower case characters, split it up into minimum number of substrings, all of which are not perfect powers.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Problem: Given string of length n, consisting of lower case characters, split it up into minimum number of substrings, all of which are not perfect powers.

A string s is a perfect power if it can be written as t^k (string t repeated k times), for $k > 1$.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Problem: Given string of length n, consisting of lower case characters, split it up into minimum number of substrings, all of which are not perfect powers.

A string s is a perfect power if it can be written as t^k (string t repeated k times), for $k > 1$.

Solution: First check if the answer is 1, by checking if s is a power.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Problem: Given string of length n, consisting of lower case characters, split it up into minimum number of substrings, all of which are not perfect powers.

A string s is a perfect power if it can be written as t^k (string t repeated k times), for $k > 1$.

Solution: First check if the answer is 1, by checking if s is a power.

Observation 1: If $s = t^k$, $|t|$ must be a divisor of n.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Problem: Given string of length n, consisting of lower case characters, split it up into minimum number of substrings, all of which are not perfect powers.

A string s is a perfect power if it can be written as t^k (string t repeated k times), for $k > 1$.

Solution: First check if the answer is 1, by checking if s is a power.

Observation 1: If $s = t^k$, $|t|$ must be a divisor of n.

Observation 2: For $n \leq 3 \cdot 10^5$, n has at most 180 divisors.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Problem: Given string of length n, consisting of lower case characters, split it up into minimum number of substrings, all of which are not perfect powers.

A string s is a perfect power if it can be written as t^k (string t repeated k times), for $k > 1$.

Solution: First check if the answer is 1, by checking if s is a power.

Observation 1: If $s = t^k$, $|t|$ must be a divisor of n.

Observation 2: For $n \leq 3 \cdot 10^5$, n has at most 180 divisors.

Algorithm: Try all divisors d of n, take $t = s[0 : d]$ and check whether $t^k = s$, $\mathcal{O}(d(n)n)$ overall.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Problem: Given string of length n, consisting of lower case characters, split it up into minimum number of substrings, all of which are not perfect powers.

A string s is a perfect power if it can be written as t^k (string t repeated k times), for $k > 1$.

Solution: First check if the answer is 1, by checking if s is a power.

Observation 1: If $s = t^k$, $|t|$ must be a divisor of n.

Observation 2: For $n \leq 3 \cdot 10^5$, n has at most 180 divisors.

Algorithm: Try all divisors d of n, take $t = s[0 : d]$ and check whether $t^k = s$, $\mathcal{O}(d(n)n)$ overall.

Observation 3: If $s =$ `aaa...aaa` , then the answer is n, because any substring with length greater than 1 is a perfect power.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Problem: Given string of length n, consisting of lower case characters, split it up into minimum number of substrings, all of which are not perfect powers.

A string s is a perfect power if it can be written as t^k (string t repeated k times), for $k > 1$.

Solution: First check if the answer is 1, by checking if s is a power.

Observation 1: If $s = t^k$, $|t|$ must be a divisor of n.

Observation 2: For $n \leq 3 \cdot 10^5$, n has at most 180 divisors.

Algorithm: Try all divisors d of n, take $t = s[0 : d]$ and check whether $t^k = s$, $\mathcal{O}(d(n)n)$ overall.

Observation 3: If $s =$ `aaa...aaa` , then the answer is n, because any substring with length greater than 1 is a perfect power.

Observation 4: If the answer is not 1 or n, the answer is 2!

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).

Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).

Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power

If $s[1 : n]$ is a perfect power, then it can be written as u^m for some string u and $m \geq 2$.



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).

Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power

If $s[1 : n]$ is a perfect power, then it can be written as u^m for some string u and $m \geq 2$.

For the string $s[1 : n]$, we know $s[i] = s[i + |t|]$ and $s[i] = s[i + |u|]$ for all i, because s is a power of t, and $s[1 : n]$ a power of u.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).

Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power

If $s[1 : n]$ is a perfect power, then it can be written as u^m for some string u and $m \geq 2$.

For the string $s[1 : n]$, we know $s[i] = s[i + |t|]$ and $s[i] = s[i + |u|]$ for all i, because s is a power of t, and $s[1 : n]$ a power of u.

Gcd calculations: $\gcd(n, n - 1) = 1$, and $|t|$ and $|u|$ are divisors of n and $n - 1$, so $\gcd(|t|, |u|) = 1$.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).

Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power

If $s[1 : n]$ is a perfect power, then it can be written as u^m for some string u and $m \geq 2$.

For the string $s[1 : n]$, we know $s[i] = s[i + |t|]$ and $s[i] = s[i + |u|]$ for all i, because s is a power of t, and $s[1 : n]$ a power of u.

Gcd calculations: $\gcd(n, n-1) = 1$, and $|t|$ and $|u|$ are divisors of n and $n-1$, so $\gcd(|t|, |u|) = 1$.

Periodicity: Because $|t| + |u| \leq n-2$, these equality constraints force s to have all equal characters.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).
Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power
If $s[1 : n]$ is a perfect power, then it can be written as u^m for some string u and $m \geq 2$.
For the string $s[1 : n]$, we know $s[i] = s[i + |t|]$ and $s[i] = s[i + |u|]$ for all i, because s is a power of t, and $s[1 : n]$ a power of u.

Gcd calculations: $\gcd(n, n - 1) = 1$, and $|t|$ and $|u|$ are divisors of n and $n - 1$, so $\gcd(|t|, |u|) = 1$.

Periodicity: Because $|t| + |u| \leq n - 2$, these equality constraints force s to have all equal characters.

Details: Standing at position i, making hops of $\pm|t|$ and $\pm|u|$, prove there always exists hop sequence to position 0.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).
Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power
If $s[1 : n]$ is a perfect power, then it can be written as u^m for some string u and $m \geq 2$.
For the string $s[1 : n]$, we know $s[i] = s[i + |t|]$ and $s[i] = s[i + |u|]$ for all i, because s is a power of t, and $s[1 : n]$ a power of u.

Gcd calculations: $\gcd(n, n - 1) = 1$, and $|t|$ and $|u|$ are divisors of n and $n - 1$, so $\gcd(|t|, |u|) = 1$.

Periodicity: Because $|t| + |u| \leq n - 2$, these equality constraints force s to have all equal characters.

Details: Standing at position i, making hops of $\pm|t|$ and $\pm|u|$, prove there always exists hop sequence to position 0.

Conclusion: If the answer is not 1 or n, then $s[1 : n]$ is not a perfect power, so $s[0] + s[1 : n]$ is a valid split.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).
Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power
If $s[1 : n]$ is a perfect power, then it can be written as u^m for some string u and $m \geq 2$.
For the string $s[1 : n]$, we know $s[i] = s[i + |t|]$ and $s[i] = s[i + |u|]$ for all i, because s is a power of t, and $s[1 : n]$ a power of u.

Gcd calculations: $\gcd(n, n - 1) = 1$, and $|t|$ and $|u|$ are divisors of n and $n - 1$, so $\gcd(|t|, |u|) = 1$.

Periodicity: Because $|t| + |u| \leq n - 2$, these equality constraints force s to have all equal characters.

Details: Standing at position i, making hops of $\pm|t|$ and $\pm|u|$, prove there always exists hop sequence to position 0.

Conclusion: If the answer is not 1 or n, then $s[1 : n]$ is not a perfect power, so $s[0] + s[1 : n]$ is a valid split.

Running time: $\mathcal{O}(nd(n))$. Results in $\approx 6 \cdot 10^7$ simple operations.

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).
Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power
If $s[1 : n]$ is a perfect power, then it can be written as u^m for some string u and $m \geq 2$.
For the string $s[1 : n]$, we know $s[i] = s[i + |t|]$ and $s[i] = s[i + |u|]$ for all i, because s is a power of t, and $s[1 : n]$ a power of u.

Gcd calculations: $\gcd(n, n - 1) = 1$, and $|t|$ and $|u|$ are divisors of n and $n - 1$, so $\gcd(|t|, |u|) = 1$.

Periodicity: Because $|t| + |u| \leq n - 2$, these equality constraints force s to have all equal characters.

Details: Standing at position i, making hops of $\pm|t|$ and $\pm|u|$, prove there always exists hop sequence to position 0.

Conclusion: If the answer is not 1 or n, then $s[1 : n]$ is not a perfect power, so $s[0] + s[1 : n]$ is a valid split.

Running time: $\mathcal{O}(nd(n))$. Results in $\approx 6 \cdot 10^7$ simple operations.

Bonus: Can you find an $\mathcal{O}(n)$ solution?

E: Endurtaka Mig

Problem author: Jeroen Op de Beek



Observation 4: If the answer is not 1 or n, the answer is 2!

Proof: Assume $s = t^k$, for $k > 1$ (otherwise answer is 1).
Take the splitting $s[0] + s[1 : n]$. $s[0]$ is a single character, so not a perfect power
If $s[1 : n]$ is a perfect power, then it can be written as u^m for some string u and $m \geq 2$.
For the string $s[1 : n]$, we know $s[i] = s[i + |t|]$ and $s[i] = s[i + |u|]$ for all i, because s is a power of t, and $s[1 : n]$ a power of u.

Gcd calculations: $\gcd(n, n - 1) = 1$, and $|t|$ and $|u|$ are divisors of n and $n - 1$, so $\gcd(|t|, |u|) = 1$.

Periodicity: Because $|t| + |u| \leq n - 2$, these equality constraints force s to have all equal characters.

Details: Standing at position i, making hops of $\pm|t|$ and $\pm|u|$, prove there always exists hop sequence to position 0.

Conclusion: If the answer is not 1 or n, then $s[1 : n]$ is not a perfect power, so $s[0] + s[1 : n]$ is a valid split.

Running time: $\mathcal{O}(nd(n))$. Results in $\approx 6 \cdot 10^7$ simple operations.

Bonus: Can you find an $\mathcal{O}(n)$ solution?

Statistics: 92 submissions, 26 accepted, 14 unknown