Ex: $s = "PXQXYXA"$, $k = 2$

$L - (\text{most occurring char freq}) \le k$

↳ for any valid substring

X - 3
A - 1
P - 1
Q - 1
Y - 1

## Sliding window + 2 ptr soln

$s = "PXQXYXA"$ (L, R)

| X | P | Q | Y | A |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |

→ freq[]

ans = 1

move the R till substring is valid. Store the freq of each char ⇒ most occurring freq.

The moment we encounter the invalid substring, we will update L

$s = "PXQXYXA"$

| X | P | Q | Y | A |
|---|---|---|---|---|
| 2 | 1 | 1 | 0 | 0 |

→ freq[]

valid { 2-1 ≤ 2, 3-1 ≤ 2, 4-2 ≤ 2

ans = 4

$s = "PXQXYXA"$, k = 2

| X | P | Q | Y |
|---|---|---|---|
| 2 | 1 | 1 | 1 |

→ freq[]

invalid → 5-2 > 2

$s = "PXQXYXA"$

| X | P | Q | Y |
|---|---|---|---|
| 2 | 1 | 1 | 1 |

→ freq[]

(++) (--)

L > ans → ans = L = 5 (length)

L = R - L + 1 (curr substring length)

freq[L]--;
L++;
R++;
freq[R]++;

| X | P | Q | Y |
|---|---|---|---|
| 3 | 0 | 1 | 1 |

→ freq[]

L - f = 5 - 3 = 2 = K → valid

R++

$s = "PXQXYXA"$

| X | A | Q | Y |
|---|---|---|---|
| 3 | 1 | 1 | 1 |

L - f = 6 - 3 > K → invalid

freq[L]--;
L++;
R++;
freq[R]++;

$s = "PXQXYXA"$

| X | A | Q | Y |
|---|---|---|---|
| 2 | 1 | 1 | 1 |

R → out of bound ↳ exit

```
if (freq[R] > maxfreq)
    maxfreq = freq[R];
```

↳ Keep track of most occurring char freq in the current substring.