Proving that Boyer Moore runs in linear time

When running the algorithm
the pattern is matched to the text until a mismatch found and then shift the pattern to the right. The goal
is to shift by the least amount of characters.

Definitions

$|\alpha|$ - period

**Periodic string** $S = \alpha \alpha \alpha \alpha \ldots (\alpha^i)$

many strings are not fully periodic

**Semi-periodic** $S = \text{suf}(\alpha) \alpha^i$

e.g.

```
TGACTGACTGACTGACTGACTG
```

**Prefix semi-periodic** $S = \alpha^i\text{pref}(\alpha)$

Every semi-periodic string is also prefix semi-periodic. A is different, but both definitions work for such a string.

Lemma:

$S = \delta\gamma = \gamma\delta \Rightarrow \delta = \alpha^i, \gamma = \alpha^j$

assume $|S| = n$ and $|\delta| > |\gamma|$

$\delta\gamma = \gamma \delta$

$\delta\gamma'\gamma = \gamma\gamma'\delta \Rightarrow \gamma'\gamma = \gamma\gamma' \Rightarrow$ by induction $\delta$ is periodic so $\gamma$ is also periodic
If P matches at positions p and p' in text and p – p' < |p|/2 then p is semi-periodic with period p' – p

Definitions

- \( t_i \): set of characters that were matched at phase i
- \( p \): suffix of pattern that contains both \( t_i \) and one more mismatched character \( |p| = |t_i| + 1 \)
- \( S_i \): # of characters that I jumped at phase i
- \( \beta \): the smallest possible period of \( \alpha \)
- \( \alpha \): \( \alpha = \beta \) – smallest \( \beta \) such that \( \alpha = \beta' \)
- \( g_{i+1} \): # of characters matched in phase \( i + 1 \) not for the first time

\(|t_i| + 1 = g_{i+1} + g'_i\)

We will prove that \( g_i < 3S_i \)
\[ \sum_{i} (g_i + g'_i) \leq m + \sum_{i} g_i \leq m + 3 \sum_{i} S_i \leq m + 3m = 4m \]

if \( S_i \geq (|t_i| + 1)/3 \) then \( g_i < 3S_i \) trivially

assume \( S_i \geq (|t_i| + 1)/3 \)

I

If \( S_i \geq (|t_i| + 1)/3 \) then \( p \& t_i \) are semi-periodic with period \( \alpha \). The proof is the same as Lemma (shifting strings)

II

At stage \( h < i \) end of \( P \) cannot coincide with boundary of \( \beta \) unit

we know that after stage \( h \) we shifted pattern somewhere. We have two possibilities:
1. Pattern matched the boundary of \( \beta \) -> clearly we could not shifted beyond \( i \) -> this option is not possible
2. Shifted such that we hit somewhere inside \( \beta \) boundary -> not possible either since \( \beta \) is the smallest possible shift and if such shift happened it would contradict that \( \beta \) is the smallest.

III

At any stage \( h < i \) “work < |\beta| \) in other words \( th overlaps t_i < |\beta| \)

\( x | \beta | \beta | \beta \) => this implies that \( \beta \) is not smallest => contradiction => at any stage our work does not overlap
IV

At stage \( h < i \) the rightmost end of pattern can only line up with the rightmost \( |\beta| - 1 \) characters of \( t_i \), or leftmost \( |\beta| \) characters of \( t_i \).
We prove that it is impossible to escape the boundaries of \( \beta \).

\[
\begin{array}{c}
\times \\
\ldots \\
\ldots
\end{array}
\]

We show that \( g_i < 3 \beta \), we know that \( \beta \leq S_i \Rightarrow g_i \leq 3S_i \)

# of characters I saw in past is bounded by shifts I do and # of shifts is bounded by \( m \) \Rightarrow \( g_i \leq m \)