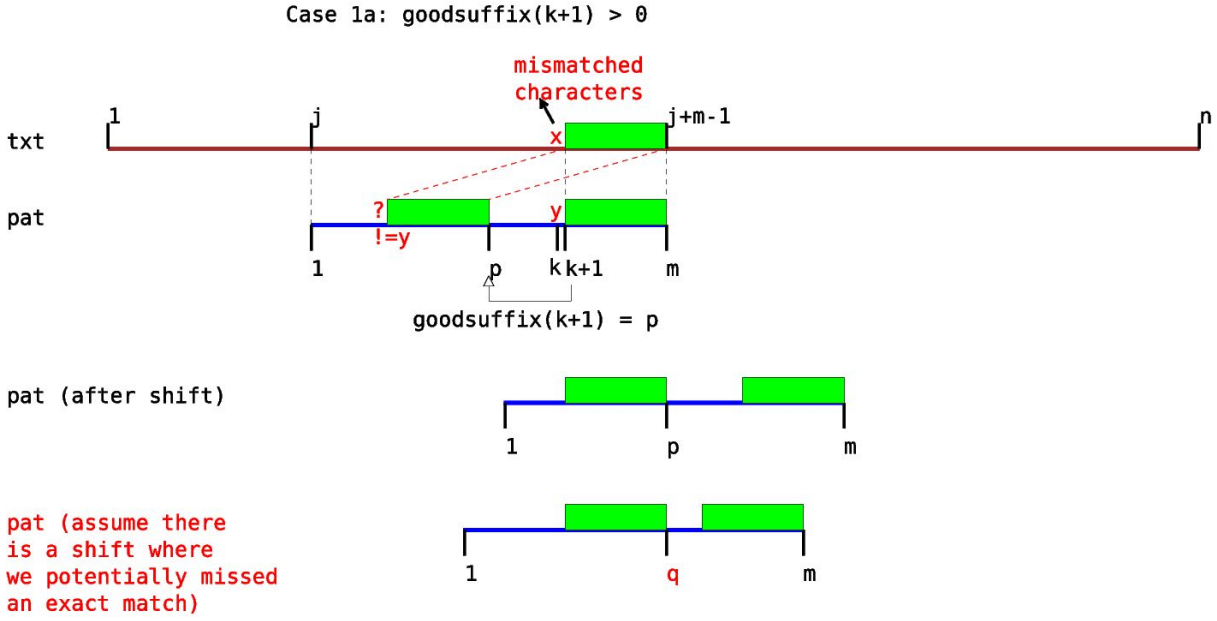
FIT3155: Week 3 Tutorial - Answer Sheet

(Scribe: Dinithi Sumanaweera)

Question 2

Prove that when a good suffix is found (see slide #37 in your lecture slides) the proposed shift-rule (on that slide) never shifts `pat` incorrectly past an occurrence in `txt`, and hence is a safe shift.

Consider the following illustration:



In the above illustration, the good suffix rule proposes a shift rightwards of pat under txt by m-goodsuffix(k+1) places.

Assume that this is not a safe shift (meaning, shifting this way we potentially miss one or more exact matches).

If that were true, as can be seen in the figure above (red highlighted shift) we arrive at a **contradiction.** The contraction is that, there is a position q (> p) such that goodsuffix(k+1) = q. However, we defined goodsuffix(k+1) = p is the end point of the **rightmost occurrence** in the pattern such that the:

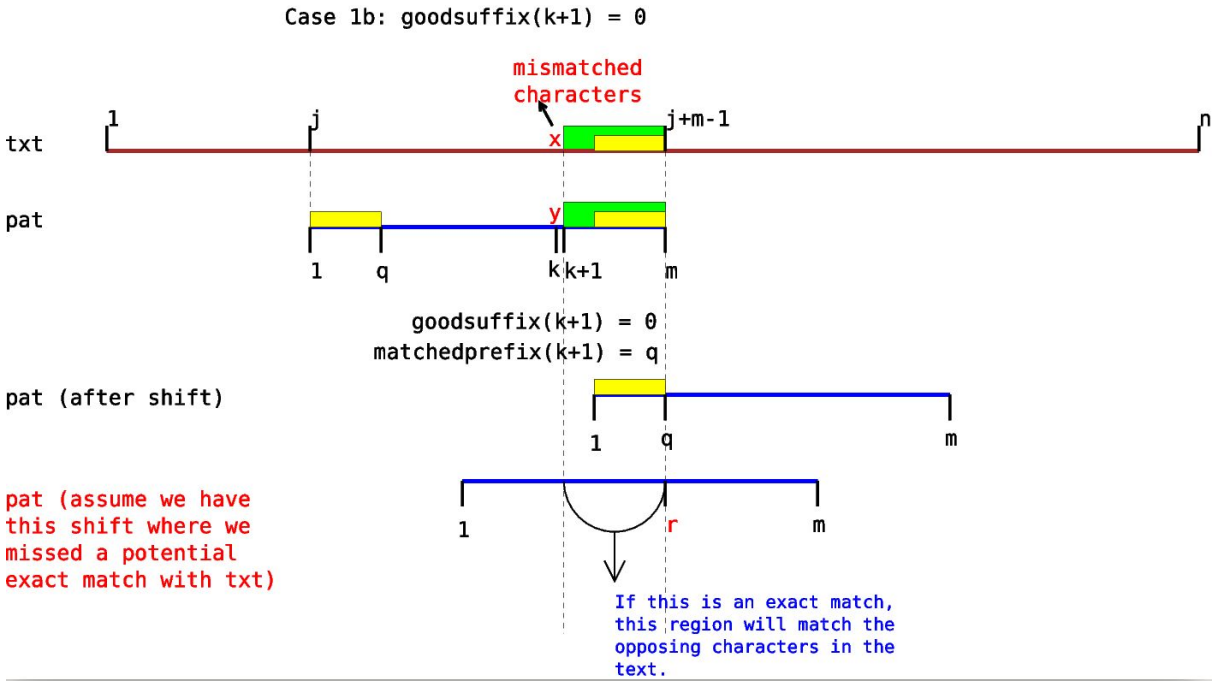
- $\mathbf{pat}[p-m+k+1 \ldots p] \equiv \mathbf{pat}[k+1 \ldots .m]$
- $\mathbf{pat}[p-m+k] \neq \mathbf{pat}[k]$.

Therefore, given this contradiction,the suggested shift rightwards by m-goodsuffix(k+1) places is safe.

Question 3

Prove that when a good suffix is NOT found (see slide #38) the proposed shift rule (on that slide) based on the precomputed `matchedprefix(.)` values, never shifts `pat` incorrectly past an occurrence in `txt`, and hence is a safe shift.

Consider the following illustration:



The suggested shift of m-matchedprefix(k+1) places is proposed only when goodsuffix(k+1) = 0

However, assume this were not a safe shift and that we potentially missed an **exact match** in between (as highlighted in red).

If this were true, it leads to a **contradiction** where goodsuffix(k+1) = r, because we have:
pat[r-m+k+1...r] = pat[k+1...m]
and
pat[r-m+k] (= txt[j+k-1]) != pat[k]

But we are handling this case only when goodsuffix(k+1) = 0, leading to a contradiction. Thus, shifting right by m-matchedprefix(k+1) placed is safe.

Question 4

When a **pat** is found in **txt**, reason why the shift rule proposed on the slide #39 is correct (and safe).

When an exact match of pat[1...m] is found at some position in the txt[1...n], the suggested shift rightwards of the pat is by m-matchedprefix(2).

Note, matchedprefix(2) gives the **longest proper** suffix of pat that matches its prefix.

Assume we missed a potential exact match where it was required to shift by only m-s places, which is LESS THAN m-matchedprefix(2) places. This implies s > matchedprefix(2).

This leads to the contradiction, as it would suggest that there is a longer proper suffix than matchedprefix(2). Hence, m-matchedprefix(2) is a safe shift.

Question 5

Refer to the slide #43 to understand the definition of $\mathbf{SP}_i$ values computed on **pat**. After this, reason why the pseudocode on slide #44 computes the $\mathbf{SP}_i$ values correctly.

Running Z-algorithm on pat[1...m], we get Z_j values for all indexes 1 < j <= m.

Any Z_j gives the length of the longest subtring of the pattern starting at position j that matches the prefix. The end point of the Z_j box is at position (say i =) j+ Z_j -1.

Clearly, multiple Z-boxes **starting** are different positions can **end** at the same position i.

SP_i by definition is length of the longest proper suffix of pat[1...i] that matches the prefix. This is same as the length of the longest Z-box whose end point is at i.

Since we want the longest such Z-box ending at each i, Line 6 of the pseudocode iterates on values of j from m down to 2, and for each j, computes the end point i on line 7, and on line 8 updates SP_i with Z_j (for the current j).

This way, Line 8 will always update SP_i with the longest (encountered so far) when multiple z-boxes (for decreasing values of j) have the same end point i.

Question 6

Refer to the slide #45. It proposes the 'KMP shift rule' of **pat** by $i - \mathbf{SP}_i$ places. Prove that this shift *never* shifts incorrectly past an occurrence of **pat** in **txt**.

Use "Proof by Contradiction" as we discussed in other questions above. If unsuccessful, come to consultation to have this clarified.