(Knuth, Morris, Pratt)

(a) Given list \( P \), find list \( L \) such that for every index \( n \) of list \( P \), \( L_n \) is the length of the longest list that is both a proper prefix and a proper suffix of \( P[0..n+1] \). Here is a program to find \( L \).

\[
A \leftarrow i := 0. \quad L := \#P*0. \quad j := 1. \quad B
\]

\[
B \leftarrow \text{if } j \geq \#P \text{ then ok else } C. \quad L \leftarrow j \rightarrow i \quad L \leftarrow j + 1. \quad B \text{ fi}
\]

\[
C \leftarrow \text{if } P_i = P_j \text{ then } i := i + 1
\]

\[
\text{else if } i = 0 \text{ then ok
\]

\[
\text{else } i := L(i-1). \quad C \text{ fi fi}
\]

Find specifications \( A \), \( B \), and \( C \) so that \( A \) is the problem and the three refinements are theorems.

(b) Given list \( S \) (subject), list \( P \) (pattern), and list \( L \) (as in part (a)), determine if \( P \) is a segment of \( S \), and if so, where it occurs. Here is a program.

\[
D \leftarrow m := 0. \quad n := 0. \quad E
\]

\[
E \leftarrow \text{if } m = \#P \text{ then } h := n - \#P \text{ else } F \text{ fi}
\]

\[
F \leftarrow \text{if } n = \#S \text{ then } h := \infty
\]

\[
\text{else if } P_m = S_n \text{ then } m := m + 1. \quad n := n + 1. \quad E
\]

\[
\text{else } G \text{ fi fi}
\]

\[
G \leftarrow \text{if } m = 0 \text{ then } n := n + 1. \quad F \text{ else } m := L(m-1). \quad G \text{ fi}
\]

Find specifications \( D \), \( E \), \( F \), and \( G \) so that \( D \) is the problem and the four refinements are theorems.