Ex. 28.2-6. Let \( P_1 = (a+b)(c-d) = ac-ad+bc-bd \)

\[
P_2 = ac
d_2 = bc
\]

\[
\therefore ac-bd = P_1 + P_2 - P_3
d + bc = P_2 + P_3
\]

only 3 multiplications.

Ex. 16.3-2

Assume there are totally \( n \) letters.

(b) The letter whose frequency is \( F_i \) (the \( i \)th Fibonacci No.),

if \( i = 1 \), then the code will contain \( n-1 \) "1"s

if \( i > 1 \), the code will contain, the first \( n-i \) "1"s and the last one is a "0"s.
Ex. 16.3-6

**HUFFMAN(C)**

- **TENARY TREE**

  \[ n \leq 1 \mid \]

  \[ Q \subseteq C \]

  for \( i = 1 \) to \( n - 1 \)

  do allocate a new node \( z \)

  \[ \text{left}[z] \leftarrow x \leftarrow \text{Extract - MIN}(Q) \]

  \[ \text{middle}[z] \leftarrow y \leftarrow \text{Extract - M2N}(Q) \]

  \[ \text{right}[z] \leftarrow t \leftarrow \text{Extract - MIN}(Q) \]

  \[ f(z) = f(x) + f(y) + f(t) \]

  return \( \text{Extract - MIN}(Q) \)

---

**Proof:**

\[
\begin{align*}
\begin{array}{c}
\text{Proof } \text{of } B(T) - B(T') \\
\text{where } B(T) = \sum x f(x) \text{ and } B(T') = \sum x f'(x)
\end{array}
\end{align*}
\]

This is to prove \( B(T) - B(T') \geq 0 \). Then we can prove \( B(T'') - B(T') \geq 0 \).

\[
B(T) = \sum x f(x)
\]

\[
B(T'') - B(T') \geq 0
\]

\[
B(T) = \sum x f(x)
\]
Ex 23.2-8

A counter-example:

\[ V_1 = \{ B, C, E, F \}, \quad V_2 = \{ A, D \}. \]

The Algorithm can never calculate a MST.

Since \( E_1 = \emptyset \)

Problem 23-4(b)

(6) Obviously, \( T \) is not a MST, a counter-example:

\[ \begin{array}{c}
A \\
\downarrow \\
B & 2 & 6 \\
\uparrow & & \\
C \\
\end{array} \]

The Algorithm may choose \( A, C \), thus total weight is \( 2 + 3 = 5 \), which is not minimal.

(b) Implementation: (similar to Kruskal's algorithm)

1. \( \text{MST} = \text{B} - \text{Maybe} (G, w) \)
2. \( T = \emptyset \)
3. for each vertex \( v \in V(G) \)
   4. do \( \text{MAKE-SET}(v) \)
5. for each edge \( (u, v) \in E \)
   6. do if \( \text{FIND-SET}(u) \neq \text{FIND-SET}(v) \)
5. then \( T \leftarrow T \cup \{u, v\} \)
6. UNION \( (u, v) \)
7. return \( T \).