1. (50 points)
Given the following pseudo-code procedure:

```plaintext
proc switch(x,y : integer) {
  proc f() : integer {
    var x : integer;
    begin
      z := x;
      x := y;
      return z;
    end proc f;
    begin
      y := f();
      end proc switch;
  }
}
```

Describe the effect of the procedure call `switch(i, a[i])` using each of the following parameter passing methods:

* call-by-value
* call-by-reference
* call-by-value-result
* call-by-name
2. (50 points)
Consider the following grammar:

\[
\begin{align*}
S & \rightarrow A \\
A & \rightarrow A + A | B + + \\
B & \rightarrow y
\end{align*}
\]

(a) Draw the parse tree for the input "y + + + y + +

(b) Complete the table below tracing an LR(1) parse of the input above.
   The "stack" column shows the stack (with the top at the right); the
   "input" column shows the unprocessed input and the "action" column
   shows which action (shift, reduce, or accept) the parser performs.

<table>
<thead>
<tr>
<th>Stack</th>
<th>Input</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>