1. An ambiguous grammar for expressions is

\[ E' \rightarrow E \downarrow \]
\[ E \rightarrow E + E \]
\[ E \rightarrow E * E \]
\[ E \rightarrow ID \]

(a) Use Earley’s algorithm to parse “ID + ID * ID” for the grammar by filling in the chart below.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>e</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>g</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>i</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>j</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>k</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(b) Draw the two accepting parse trees and identify that there are ambiguities in the grammar.
What are the two ways to reach the accept state?

**Answer:** There are two ways to reach \( E' \rightarrow E \downarrow, 0 \):

\[(ID + ID) * ID, (b4) \rightarrow (c5)\]

and

\[ID + (ID * ID), (a4) \rightarrow (b5)\]
2. Consider the following grammar:

\[ Y \rightarrow a \ c \]
\[ X \rightarrow a \ X \ | \ a \ X \ b \ X \ | \ c \]

(a) Come up with two handles for the string ‘acbc’ which exhibit a shift/reduce conflict.

**Answer:** \([X : aX \bullet, b] \) and \([X : aX \bullet bX, b]\).

(b) Come up with two handles for the string ‘ac’ which exhibit a reduce/reduce conflict.

**Answer:** \([Y : ace, \] \) and \([X : aX \bullet, -]\).

(c) Remove \(Y\) from the grammar. Introduce a precedence rule which forces the parser to shift whenever it encounters a shift/reduce conflict.

**Answer:** Give \(b\) precedence by making it associate to the right. In Bison syntax, the appropriate declaration is \(\%\text{right} \ 'b'\).

(d) Repeat the previous question, but force the parser to reduce whenever it encounters a shift/reduce conflict.

**Answer:** Introduce \(\%\text{left} \ 'b'\). Remember, the rule is: only shift to resolve a shift/reduce conflict if (1) the next token has higher precedence than the candidate production OR (2) the next token has the same precedence but associates to the right.

3. Consider the following Python program:

```python
x = 1
def f(y):
    return lambda: x + y
fun = f(3)
y = 2
print(fun())
```

(a) Using static (i.e lexical, or ‘normal’) scoping, what does this print?

**Answer:** 4.

(b) Using dynamic scoping, what does this print?

**Answer:** 3.