INSTRUCTIONS

- You have 10 minutes to complete this quiz.
- The exam is closed book, closed notes, closed computer, closed calculator.
- Mark your answers on the exam itself. We will not grade answers written on scratch paper.
- For multiple choice questions, fill in each option or choice completely.
  - □ means mark all options that apply
  - ○ means mark a single choice

<table>
<thead>
<tr>
<th>Last name</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>First name</td>
<td></td>
</tr>
<tr>
<td>Student ID number</td>
<td></td>
</tr>
<tr>
<td>CalCentral email (@berkeley.edu)</td>
<td></td>
</tr>
<tr>
<td>Discussion Section</td>
<td>___ ___ ___</td>
</tr>
</tbody>
</table>

All the work on this exam is my own. (please sign)

0. Your thoughts? How are you feeling this week?
1. **Yes, No, but Sometimes Maybe?**

Fill in the environment diagram that results from executing the code below until the entire program is finished, an error occurs, or all frames are filled. *You may not need to use all of the spaces or frames.*

A complete answer will:

- Add all missing names and parent annotations to all local frames.
- Add all missing values created or referenced during execution.
- Show the return value for each local frame.

*You must list all bindings in the order they first appear in the frame.*

```python
def yes(no):
    yes = 'no'
    return no
no = 'no'
def no(no):
    return no + yes(no)
yes = yes(no)('ok')
```

---

**Environment Diagram**

**Global Frame**
- **yes**: 'okok'
- **no**: 'okok'

**Frame 1 (f1):**
- **yes**
- **no**
- **Return Value**: 'ok'

**Frame 2 (f2):**
- **yes**
- **no**
- **Return Value**: 'no'

**Frame 3 (f3):**
- **no**
- **Return Value**: 'okok'

**Frame 4 (f4):**
- **yes**
- **Return Value**: 'ok'

---

**Function Call Flow:**
- `func yes(no) [parent=Global]`
- `func no(no) [parent=Global]`