CS 160
CS Orientation
Planning Your Solution
Chap. 3
Sequential Logic Structure

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Flowchart</th>
<th>Pseudocode</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Instruction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Instruction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Instruction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. Instruction</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- get shape
- get color
- time of the year
- get size
- texture
- if (shape == 1)
  - if (sternated)
    - ask if they aren't leaf id
Python Sequential Logic

```python
print(" * ")
print(" *** ")
print(" ***** ")
print("*******")
```
Decision Logic Structure

5. If <decision> then
   Instruction
else
   Instruction
6. : 

```plaintext
if (shape == "serrated")
   Instruction
else
   Instruction
endif
```
x=1
if(x==1):
    print(" * ")
if(x==3):
    print(" *** ")
if(x==5):
    print(" ***** ")
else:
    print("*******")
Loop Logic Structure

5. Loop
   Instruction
   Instruction
   Instruction
   Until <logical expression>

6. Instruction
   Instruction
   Instruction
   Until <logical expression>

while continue

Oregon State University
Python Loop Logic

```
for x in range(7):
    print("*", end="")

OR

x=1
while(x<=7):
    print("*", end="")
    x+=1
```

# Oregon State University
Print Triangle w/ Ifs from student

```python
if(1):    #This is the same as if(1==1) or if(True)
    print("    *")
if(1):
    print("   *** ")
if(-1):  #Any number other than 0 is true
    print("  ***** ")
if(0):
    print("********")
```
Print Triangle w/ while from student

```python
tri = 0  # Create variable to hold which part of the triangle to print

while tri < 4:
    if tri == 0:
        print("  *")
        tri += 1  # Same as tri=tri+1 to increment the variable
    elif tri == 1:
        print(" *** ")
        tri += 1
    elif tri == 2:
        print(" ***** ")
        tri += 1
    elif tri == 3:
        print(" ********")
        tri += 1
```
Print Triangle w/ loops from me😊

```python
#This would be my approach, so I can make bigger triangles quickly
#without changing this number everywhere in my program.
MAX_TRI = 8

for x in range(1, MAX_TRI, 2):
    for sp in range((MAX_TRI-x)//2):
        print(" ", end="")
    for sp in range(x):
        print("*", end="")
    for sp in range((MAX_TRI-x)//2):
        print(" ", end="")
    print(x)
```