• If you need more space to answer a question than we give you, you may use the additional blank sheet of paper attached to your exam. Make sure that we know where to look for your answer.

• Read each question carefully and make sure that you answer everything asked of you. Write legibly so that we can read your solutions. Please do not write anything in red.

• We suggest that for solutions that require you to write Python code, you include comments. They will help your grader understand what you intend, which can help you get partial credit.

• You have until noon to complete the exams
<table>
<thead>
<tr>
<th>Question</th>
<th>Value</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>
Question 1 30 points

For each part of this question, consider the Python code shown. What is printed when this code is run (there are no errors in any of these parts)?

(a) (5 points)

```python
c = 1
while c >= 0 and c <= 2:
    print(2*c)
    c = c + 1
```

SOLUTION: 2, 4

(b) (5 points)

```python
for i in range(0,3):
    if i % 2 == 0:
        for j in range(0,3):
            print(i,j)
```

SOLUTION:

0 0
0 1
0 2
2 0
2 1
2 2
(c) (5 points)

```python
def mystery(a,b,c,d):
    if a == b:
        return c
    else:
        return d
```

```python
print( mystery("a", "b", 1, 2) + 1)
```

SOLUTION: 3

(d) (5 points)

```python
X = ["a", "b", "c"]
Y = X
X[1] = [1, 2, 3]
print(Y)
```

SOLUTION: ['a', [1, 2, 3], 'c']
(e) (5 points)

```python
A = [2, 3, 4]
B = ['a', 'b', 'c', 'd', 'e']
```

SOLUTION: cde

(f) (5 points)

```python
def mystery(a, b):
    return a + b

print( mystery("one plus two ","equals ") + str(mystery(1,2)) )
```

SOLUTION: one plus two equals 3
Question 2 10 points

Write a Python function `is_equal` that takes as input two parameters and returns `True` if the two parameters are equal and `False` otherwise. The body of your function can consist of only a single line.

SOLUTION:

```python
def is_equal(x, y):
    return x == y
```
Question 3  
10 points

Write a Python function `count_up` that takes as input two integers $x$ and $y$ and prints the values $x$, $x+1$, $\ldots$, $y$. For example calling `count_up(3,7)` will yield the following output:

3
4
5
6
7

Your function must use a `for` loop to print these values. You can assume that $x < y$.

SOLUTION

```python
def count_up(x, y):
    for i in range(x, y + 1):
        print(i)
```
Question 4 10 points

Write a Python function `count_down` that takes as input two integers `x` and `y` and prints the values `y, y-1, ..., x`. For example calling `count_down(3,7)` will yield the following output:

```
7
6
5
4
3
```

Your function must use a `while` loop to print these values. You can assume that `x < y`.

SOLUTION

```python
def count_down(x, y):
    i = y
    while i >= x:
        print i
        i = i - 1
```
Question 5          10 points

Write a Python function `howlong` that takes as input a string and print the string “very short” if the length of the input is 1, prints “short” if the length of the string is greater than 1 and less than or equal to 5, prints “long” if the length of the string is greater than 5 and less than or equal to 10, and prints “very long” otherwise. Your solution cannot use the logical operators `and`, `or`, `not`. You can assume that the length of the input is greater than or equal to 1.

SOLUTION

def howlong(s):
    if len(s) == 1:
        print( "very short" )
    elif len(s) <= 5:
        print( "short" )
    elif len(s) <= 10:
        print( "long" )
    else:
        print( "very long" )
Question 6  
15 points

Consider a list where each element is a list (e.g., [[1,2,3], [4,5,6], [7,8,9]]). Write a Python function addlist that takes as input such a list X along with a single list y (e.g., [1,-1,2]). Your function should add, element-wise, the list y to each list in X. For example, in the above example, the value of X after calling your function would be [[1+1,2-1,3+2], [4+1,5-1,6+2], [7+1,8-1,9+2]]. Your solution must work for an arbitrary number of elements in the list X and an arbitrary number of elements in the list y. You can assume that each element of X will always be of the same length as y.

SOLUTION

```python
def addlist(X,y):
    for i in range( 0, len(X) ):
        for j in range( 0, len(X[i]) ):
            X[i][j] = X[i][j] + y[j]

X = [ [1,2,3], [4,5,6], [7,8,9] ]
y = [1,-1,2]
addlist(X,y)
print( X )
```
Question 7  
15 points

Write a Python function `index` that takes as input a list `L` of strings and a string `S`. If `S` is one of the elements in the list, then your function should return the index of the first occurrence of `S` in the list. For example the call `index(["ant ", "bee", "cat"], "cat")` should return 2. If the string `S` is not in the list, then your function should return -1. Your solution must use either a for or while loop and work for an arbitrary number of elements in `L`.

SOLUTION

```python
def index( L, S ):
    i = 0
    for l in L:
        if( l == S ):
            return i
        i = i + 1
    return -1
```