EXAMINATION INSTRUCTIONS

This examination has 9 pages. If your copy is missing a page, let one of the course staff know. Before starting this test, students should put his or her SUNY/UB card out on the desk so that course staff can check attendance.

This examination has 8 questions. Answer all questions.

You have 60 minutes to complete this examination. Use your time accordingly.

READ AND OBSERVE THE FOLLOWING RULES:

► Names are pre-printed on the exam booklets. Ensure that you have YOUR exam.
► Sign, using your usual signature, in the space provided on the back cover.
► All of your writing must be handed in. This booklet must not be torn or mutilated in any way and cannot leave the examination room.
► For your answers to earn credit, you must show all of your work in arriving at an answer. Partial credit will be earned as appropriate.
► Students can talk to proctors only in cases of supposed errors or question ambiguity. Proctors will not answer any other questions during this exam.
► Candidates guilty of any dishonest practices will be immediately dismissed from the examination and shall be liable for disciplinary action. Examples of dishonest practices include:
  ♦ Having books, papers, calculators, computers, cellphones, or other memory aid or electronic device out without explicit authorization by the proctors.
  ♦ Speaking or communicating with other students.
  ♦ Purposely exposing their work where other students could view it. Claims that the exposure was accidental or due to forgetfulness will not be accepted.

--------------------------- DO NOT WRITE BELOW THIS LINE! ---------------------------

<table>
<thead>
<tr>
<th>MODULE 1</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Subtotal</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/20</td>
<td>/20</td>
<td>/20</td>
<td>/20</td>
<td>/80</td>
</tr>
<tr>
<td>MODULE 2</td>
<td>Q1</td>
<td>Q2</td>
<td>Q3</td>
<td>Q4</td>
<td>Subtotal</td>
</tr>
<tr>
<td>2</td>
<td>/20</td>
<td>/20</td>
<td>/20</td>
<td>/20</td>
<td>/80</td>
</tr>
</tbody>
</table>

Total /160
Module 1 - Question 1 [20 points, 4 points each]
The code given below is syntactically correct.

Circle, and identify by number, **EXACTLY 1** example of each of the following items in the code below. If the code below does not contain an example of an item, write “no example” next to that item in the list. To show you how I want the question answered, the first one is done for you.

0. numeric literal  
1. parameter listing  
2. variable whose value is a **str** no example  
3. **str** literal. no example  
4. function call  
5. boolean expression

```
def harmonicMean(lst):
    product = 1
    for val in lst:
        if val > 0:
            product = product * val
        elif product < 0:
            product = product * val
    count = len(lst)
    retVal = product ** (1 / count)
    return retVal
```
Module 1 - Question 2  [20 points total, 10 each part]

Study the following code, then answer the question which follows.

```python
def module1question2(param) :
    retVal = ''
    if param > 62 :
        return "*"
    if param > 35 :
        retVal = retVal + "upper "
    elif param > 9:
        retVal = retVal + "lower "
    elif param > 32 :
        retVal = retVal + 'Z'
    elif param < 10 :
        retVal = retVal + str(param)
    else :
        retVal = retVal + '+'
    return retVal
```

[10 points] What does the following statement print?
```python
print(module1question2(22))
```
Write your answer below:
```
lower+
```

[10 points] Write the statement that calls `module1question2` and results in the function returning "lowerZ". Write your answer below:
```python
module1question2(x), where x is 33, 34, or 35
```
Module 1 - Question 3  [20 points total]

Read this description of how to convert a question score into a traditional grade:

The question contained 10 separate parts. The minimum score for each of the parts is -4 and the maximum score for each part is 4. This created question scores ranging from a minimum of -40 and a maximum of 40. Grades are traditionally reported as a number between 0 and 100, however. This conversion requires two separate steps. First, the score is adjusted to start at 0 by adding 40. Then the range is adjusted to be from 0 to 100 by multiplying the adjusted score by 1.25.

Define a Python function named `questGrade` which converts the question score to a traditional grade, as explained above. Your function should take the question score as its only input and return the traditional grade to which it is equal.

Write your answer below:

```python
def questGrade(score) :
    adjustedScore = score + 40
    grade = adjustedScore * 1.25
    return grade
```
Module 1 - Question 4 [20 points total]

Consider a function named \texttt{schooling} which will be called with one argument (you can assume that it will be called only with an \texttt{int} value representing a student’s age). This function will compute and return a \texttt{str} representing the type of school that student will most likely attend, according to this table:

<table>
<thead>
<tr>
<th>Age</th>
<th>School</th>
</tr>
</thead>
<tbody>
<tr>
<td>\geq 18</td>
<td>University</td>
</tr>
<tr>
<td>\geq 14 and \lt 18</td>
<td>High School</td>
</tr>
<tr>
<td>\geq 12 and \lt 14</td>
<td>Junior High</td>
</tr>
<tr>
<td>\geq 9 and \lt 12</td>
<td>Middle</td>
</tr>
<tr>
<td>\lt 8</td>
<td>Elementary</td>
</tr>
</tbody>
</table>

Describe briefly what a test case is: [ 8 points total ]

[ ] 8 points: perfect         [ ] 3 points: clearly wrong, some correct elements
[ ] 5 points: essentially correct but with small mistakes [ ] 0 points: for anything else

\textbf{A test case is a possible input together with the corresponding correct output.}

Give two distinct test cases: [ 12 points total, 6 points each ]

[ ] 6 points: perfect         [ ] 2 points: clearly wrong, some correct elements
[ ] 4 points: essentially correct but with small mistakes [ ] 0 points: for anything else

Test case #1:

\texttt{schooling(18)} \rightarrow \texttt{‘University’}

Test case #2:

\texttt{schooling(2)} \rightarrow \texttt{‘Elementary’}
Module 2 - Question 1 [20 points, 4 points each]
The code given below is correct: it compiles without errors. I have added some extra spacing to make this question easier to answer.

Circle, and identify by number, one *and only one* example of each of the following items in the code below. If you believe no example exists, write “no example” next to that item in the list. To show you how I want the question answered, the first one is done for you.

0. numeric literal
1. conditional statement (entire statement)
2. looping statement (entire statement)
3. empty dictionary literal *no example*
4. loop header
5. use of a list index *no example*

```python
def harmonicMean(lst):
    product = 1
    for val in lst:
        if val > 0:
            product = product * val
        elif product < 0:
            product = product * val
    count = len(lst)
    retVal = product ** (1 / count)
    return retVal
```

4

3

1

0

2

0

1

4

2

1
Module 2 - Question 2 [20 points total]

Consider a JavaScript function named `prodNegative` which has one parameter. The parameter is an array of numbers. The function calculates the product of all the numbers in the array that are negative (smaller than 0). If all of the numbers in the array are 0 or positive, the function can return 1.

Part 1 [1 point]
What value does `prodNegative( [ 0, 0 ] )` return?

1

Part 2 [1 point]
What value does `prodNegative( [ -1, 2, -9 ] )` return?

9 (= -1 * -9)

Part 3 [1 point]
What value does `prodNegative( [ -1, -2, -10 ] )` return?

-20 (= -1 * -2 * -10)

Part 4 [17 points]

[ ] 17 points: perfect
[ ] 6 points: clearly wrong, some correct elements
[ ] 12 points: essentially correct but with small mistakes [ ] 0 points: for anything else

Define, in JavaScript, the function `prodNegative`:

```javascript
function prodNegative(arr) {
    var retVal = 1;
    for (var ent of arr) {
        if (ent < 0) {
            retVal = retVal * ent
        }
    }
    return retVal;
}
```
Module 2 - Question 3 [20 points total]

[ ] 20 points: perfect
[ ] 7 points: clearly wrong, some correct elements
[ ] 14 points: essentially correct but with small mistakes
[ ] 0 points: for anything else

Define a Python function named `skipList` with two parameters. The first parameter for this function is a List of `str` and the second parameter is an `int`. This function will return a new List containing every \( n \)th entry from the first parameter, where \( n \) is the value of the second parameter. The original List must not be modified.

For example,

\[
skipList(['sue', 'amy', 'bob'], 2)
\]

would return a new List with: ['sue', 'bob'], whereas

\[
skipList(['every', 'third', 'word', 'example', 'is', 'hard', '.'], 3)
\]

must return a new List containing: ['every', 'example', '.']

Write your answer below:

```python
def skipList(lst, n):
    retVal = []
    for idx in range(0, len(lst), n):
        retVal.append(lst[idx])
    return retVal
```
Module 2 - Question 4 [20 points total]
[ ] 20 points: perfect
[ ] 7 points: clearly wrong, some correct elements
[ ] 14 points: essentially correct but with small mistakes
[ ] 0 points: for anything else

Study the following code:

```python
import csv

def hidden_words(filename):
    with open(filename, "r") as f:
        reader = csv.reader(f)
        for line in reader:
            if line[1] == "P":
                print(line[0] + line[2])
        print("Done")

hidden_words("steg.csv")
```

Give possible contents for the file `steg.csv` which would cause the above to print:

```
If you
    can read
this I’m
Done
```

Give your answer below:

```
If ,P , you
anything , skipped , why not , more allowed
can ,P , read
this ,P , I’m
```