EXAMINATION INSTRUCTIONS

This examination has 9 pages. Check that you have a complete paper. Each candidate should be prepared to produce, upon request, his or her SUNY/UB card.

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 must not be taken from the examination room.
► Show all of your work in arriving at an answer, unless instructed otherwise. Partial credit will be awarded as appropriate.
► Candidates are not permitted to ask questions of the invigilators, except in cases of supposed errors or ambiguities in examination questions.
► CAUTION – Candidates guilty of any of the following, or similar, dishonest practices shall be immediately dismissed from the examination and shall be liable to disciplinary action.
  ♦ Making use of any books, papers or memoranda, calculators or computers, audio or visual cassette players, or other memory aid devices, other than those explicitly authorized by the examiners.
  ♦ Speaking or communicating with other candidates.
  ♦ Purposely exposing written papers to the view of other candidates. The plea of accident or forgetfulness shall not be received.

--------------------------- 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></td>
<td>/20</td>
<td>/20</td>
<td>/20</td>
<td>/20</td>
<td>/80</td>
</tr>
</tbody>
</table>

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

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

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. parameter name
2. a string literal
3. assignment statement
4. a function call – no example
5. a Boolean expression

```javascript
function odd_a_average (names) {
    var sum = 0;
    var count = 0;
    for (var k = 0; k < names.length; k = k + 2) {
        name = names[k];
        if (name[0] == "a") {
            sum = sum + name.length;
            count = count + 1;
        }
    }
    return sum / count;
}
```

1 (but could be identified in code below too)
2
3 (there are 5 other assignment stmts above)
Module 1 - Question 2 [20 points total, 10 each part]

10 points: perfect

[ ] 3 points: clearly wrong, some correct elements

[ ] 7 points: essentially correct but with small mistakes

[ ] 0 points: for anything else

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

```javascript
function whatDoesThisDo(x) {
    result = "Serenity";
    if (x > 450) {
        return "Mal";
    }
    if (x < 300) {
        result = result + " ship";
    } else if (x < 430) {
        result = result + " firefly";
    }
    if (x < 200) {
        result = result + " class";
    } else if (x > 425) {
        result = result + " transport";
    } else {
        result = result + " home";
    }
    return result;
}
```

[10 points] What does the following statement print?

```javascript
console.log(whatDoesThisDo(300));
```

Write your answer below:

Serenity firefly home

[10 points] Give a value for x so that when whatDoesThisDo(x) is called the function returns "Serenity firefly transport". Write your answer below:

426 (any value from 426 to 429, inclusive, is acceptable)
Module 1 - 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

Read this description of how to compute the score on a fictional test:

The exam consists of 20 questions. The minimum score for each question is -4, and the maximum score is +4. The possible raw score range is therefore -80 to 80. However, the score will be reported as a number between 0 and 100. The conversion happens in two steps. First, the raw score is translated to the range 0 to 160 by adding 80 to the score. Next, the resulting translated score is scaled to the range 0 to 100 by dividing by 1.6, to get the adjusted score.

Define a Python function named ‘adjustedScore’ which converts a raw test score to a translated and scaled test score, as explained above. The function must accept a test score in raw form and return the corresponding translated and scaled score.

Write your answer below:

```python
def adjustedScore(x):
    return (x+80) / 1.6
```
Module 1 - Question 4 [20 points total]

Consider a function named ‘courseGrade’ which will be called with one argument (you can assume that it will be called only with an integer value representing a course grade) that computes a string value representing the corresponding letter grade, according to this table:

<table>
<thead>
<tr>
<th>Course grade (%)</th>
<th>Course grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; 95</td>
<td>Exceeds expectations</td>
</tr>
<tr>
<td>&gt; 80 and &lt;= 95</td>
<td>Meets expectations</td>
</tr>
<tr>
<td>&gt; 70 and &lt;= 80</td>
<td>Marginal</td>
</tr>
<tr>
<td>&gt; 65 and &lt;= 70</td>
<td>Acceptable</td>
</tr>
<tr>
<td>&lt;= 65</td>
<td>Not meeting standards</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

A test case is a set of possible inputs 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:

Input: 68, output: "Acceptable"

Test case #2:

Input: -34, output "Not meeting standards"
Module 2 - Question 1 [20 points, 4 points each]
The code given below is correct: it compiles without errors. I have added some extra spaceing 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. a relational operator
4. the body of a for statement
5. an array access

```javascript
function odd_a_average(names) {
    var sum = 0;
    var count = 0;
    for (var k = 0; k < names.length; k = k + 2) {
        name = names[k];
        if (name[0] == "a") {
            sum = sum + name.length;
            count = count + 1;
        }
    }
    return sum / count;
}
```

The entire statement MUST be identified; the inner loop is also an acceptable answer.

The opening and closing braces should be included, but if they are not (or if only one is) we'll accept that too.
Module 2 - Question 2 [20 points total]

Consider a Python function named ‘longInList’ which has one parameter. The function will be called with a list of strings and will return a new list of all the strings which have length at least 5, in the same relative order that they appeared in the original list.

Part 1 [1 point]
What value does longInList([ "kitten", "puppy", "dog" ]) return?

['kitten', 'puppy']

Part 2 [1 point]
What value does longInList([ "cat", "pig", "dog", ] ) return?

[]

Part 3 [1 point]
What value does longInList([ ] ) return?

[]

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 Python, the function longInList:

```python
def longInList(x):
    answer = []
    for v in x:
        if len(v) >= 5:
            answer.append(v)
    return answer
```
Module 2 - Question 3 [20 points total]
[ ] 20 points: perfect
[ ] 14 points: essentially correct but with small mistakes
[ ] 7 points: clearly wrong, some correct elements
[ ] 0 points: for anything else

Define a JavaScript function named zip with two parameters. Assume the function will be called with two equal-length lists. Define the function so it returns a new list containing all the strings from the original lists, in the same relative order. The original list must not be modified.

For example,
zip(["cat", "dog"], [1, 2]) must return [ 'cat', 1, 'dog', 2 ],
zip([1, 3, 5, 7], [0, 2, 4, 6])) must return [ 1, 0, 3, 2, 5, 4, 7, 6 ], and
zip([], []) must return [].

Write your answer below:

```javascript
function zip(x,y) {
    result = [];
    for (k in x) {
        result.push(x[k]);
        result.push(y[k]);
    }
    return result;
}
```
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 mystery(filename):
    with open(filename, "r", newline='') as f:
        reader = csv.reader(f)
        for line in reader:
            print(line[2] + " -> " + line[4])
        print("Finished")

mystery("f.csv")
```

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

- Kaylee -> Frye
- Malcolm -> Reynolds
- River -> Tam
- Finished

Give your answer below:

Here's one possible solution. There are many others:

```
0,0,"Kaylee",0,"Frye"
0,0,"Malcolm",0,"Reynolds"
0,0,"River",0,"Tam"
```