

This Exam is being given under the guidelines of the **Honor Code**. You are expected to respect those guidelines and to report those who do not. Answer the questions in the spaces provided. If you run out of room for an answer, continue on the back of the page. There are 8 questions for a total of 100 points.

Name: _____

1. Suppose you are writing an application to keep the payroll for some company. Your application consists of at least three classes: **Person**, **Manager**, and **Worker**. A **Person** has a name and year of birth. A **Worker** has a department and a monthly salary. A **Manager** has a department and a salary and a bonus equal to 10% of their salary.
 - (a) (10 points) Describe the inheritance hierarchy you will use in this application. Indicate in your description where the instance variables belong.
 - (b) (10 points) Implement the classes on your inheritance hierarchy. In particular you should write an efficient constructor for each class, and an `__str__` method for each class, and a method that returns the total yearly compensation.

2. (5 points) Convert the following expression to its fully parenthesized form.

$$A - B * C / D + E + F$$

3. (5 points) Convert the following expression to its Postfix form.

$$A / B * C + D$$

4. (5 points) Show how the stack is used in evaluating the following postfix expression.

$$7\ 13\ *\ 12\ 8\ -\ 4\ +\ 10\ *$$

5. (5 points) Convert the following expression to prefix notation.

$$A * B / (C - D) + (E * F)$$

6. (20 points) For this question you must implement a `Stack` class. However you may **not** use a list in your implementation. You **must** use two `Queues`. Implement the methods `__init__`, `push`, `pop`, and `isempty`.

7. Write a recursive function that finds the minimum value in a list of numbers. The function must accept a list

(a) (5 points) What is the base case?

(b) (5 points) What is your ‘recursive step?’

(c) (10 points) Write the function `findMin(l)`

8. (20 points) Given the following mystery function. Trace the calls and show what it prints out when called as shown.

```
1      def mystery(s):
2          if s == "":
3              return [s]
4          else:
5              ans = []
6              for w in mystery(s[1:]):
7                  for pos in range(len(w) + 1):
8                      ans.append(w[:pos]+s[0]+w[pos:])
9              return ans
10     mystery('top')
```