Write your name at the top of this page. Do not write your name on any other pages.

You have 70 minutes. Notes, books, computers, etc. are forbidden.

Show your work. Incorrect or incomplete answers with good work often earn partial credit. Correct answers with insufficient explanation might not earn full credit.

If you are asked to write code but you do not know the exact Python required, then try to write code that is approximately correct. Feel free to describe your idea in English as well. Be precise enough that I cannot misinterpret your solution.

Good luck!

```
class Employee(object):

    def __init__(self, birthYear, salary):
        self.birthYear = birthYear
        self.salary = salary
        self.startNewYear()

    def getBirthYear(self):
        return self.birthYear

    def getAge(self):
        return self.age

    def raiseSalary(self, increase):
        self.salary *= 1.0 + increase

    def payMonth(self):
        self.payThisYear += self.salary / 12.0

    def payBonus(self, bonus):
        self.payThisYear += bonus

    def startNewYear(self):
        self.payThisYear = 0.0
        yearsSince1970 = time.time() / (60 * 60 * 24 * 365.5)
        self.age = yearsSince1970 + (1970 - self.birthYear)
```

```
    def getSalary(self):
        return self.salary

    def getPayThisYear(self):
        return self.payThisYear
```

A0. Write one line of code, that demonstrates how the `reduce` function of the `functools` module can be used on a list of numbers. Show both your command and Python's response.

A1. Imagine that Python did not come with `reduce`. Write it from scratch.

B0. Design a new 3×3 edge-detection kernel for use with `imageConvolved`. Your kernel should produce large numbers at pixels where (A) there is an edge running diagonally from top-right to bottom-left, and (B) the image is darker below the edge than above it.

B1. For an $n \times n$ image convolved by a $k \times k$ kernel, what is the running time of `imageConvolved`? State your answer using $\mathcal{O}$, n, and k, and briefly explain.

C0. We have studied two GCD algorithms: trial division (`gcd`) and the Euclidean algorithm (`euc`). Sketch a graph to illustrate how both of their running times depend on the number n of bits in the inputs. Annotate your graph with explanatory comments, probably involving $\mathcal{O}$.

C1. Imagine that you are describing this topic to your relative, who has never studied computer science. Which algorithm is better, or does it not matter much? And why are GCDs important?

D. Write a recursive function to compute the nth Fibonacci number F_n. ($F_0 = 0$ and $F_1 = 1$.)

E. The following command produces an error. Give the error message, including the debugging information. Explain how Python produces that error message. A picture may help.

```
mergeSort([5, 2, 7, 6, 3, 0, '1', 4])
```

F0. Using the `Employee` class on the front page of this exam, make a new employee, pay her a month of salary, query her pay so far this year, and show Python's response.

F1. Now the company maintains a retirement account for each employee. The total value of this account is kept in the employee's record. Every month the company adds money to this account when the employee's salary is paid. The amount added varies from employee to employee. For tax purposes, these payments are included in the employee's pay-this-year figure. Write any new or changed methods necessitated by this feature.