Department of Electrical Engineering and Computer Science

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.095: Introduction to Computer Science and Programming

Quiz I

In order to receive credit you must answer the question as precisely as possible. You have 50 minutes to finish this quiz.

Write your name on this cover sheet AND at the bottom of each page of this booklet.

If you find a question ambiguous, consult the staff, and be sure to write down any assumptions you make. Be neat. If we can't understand your answer, we can't give you credit!

Some questions may be much harder than others. Read them all through first and attack them in the order that allows you to make the most progress.

1 (xx/24)	2 (xx/15)	3 (xx/10)	4 (xx/21)	5 (xx/15)	6 (xx/15)	Total (xx/100)

Name: *Alyssa P. Hacker*

I True or False [24 points]

Are each of the following True or False:

1. **[4 points]:** In Python, strings are mutable. ***False***

2. **[4 points]:** In Python, dictionaries are mutable. ***True***

3. **[4 points]:** Greedy algorithms are often used to solve optimization problems. ***True***

4. **[4 points]:** Divide and conquer algorithms are always based on a heuristic. ***False***

5. **[4 points]:** A program that uses an O(n^2) algorithm will always take longer to run than a program that uses an O($n \log n$) algorithm. ***False***

6. **[4 points]:** Brute force solutions should never be used. ***False***

Name: *Alyssa P. Hacker*

II Time Complexity [15 points]

Characterize the efficiency, using "big O" notation, of the following function:

```
def do_something(n):
    # n: int
    # a: dictionary (int -> int)
    a = {}
    for i in range(0, n):
        a[i] = i
    for i in range(0, n):
        for j in range(0, n):
            print a[i] * a[j]
```

Assume `print` is $O(1)$.

$\boxed{O(n^2)}$

Look at the nested loop.

III Graphs [10 points]

Define each of the following:

1. [5 points]: Directed graph

Any reasonable definition.

For example: a directed graph is a set of nodes N and a set of edges E, where each edge is an ordered pair $(n1, n2)$ and $n1, n2 \in N$.

2. [5 points]: Cycle (in a graph)

Any reasonable definition.

For example: a cycle is a sequence of edges $(n'_1, n'_2), ..., (n_1, n_2)$, where $n'_1 = n_2$.

IV Factorial [21 points]

Write a specification and implementation for a Python function that computes factorial. It should not use recursion.

```
def factorial(n):
    """
    Computes and returns the factorial of n.
    The factorial of 0 is 1.

    n: int >= 0
    returns: int
    """
    # ans: int (the answer)
    ans = 1
    while n > 1:
        ans = ans * n
        n -= 1
    return ans
```

Important parts of the answer: correctly computes factorial; returns the result properly; specification makes sense; covers excceptional cases (e.g., input must be a natural number).

V Type Checking [15 points]

Python does not associate a type with a variable, e.g., the variable `x` can refer to an integer and then a float within the same scope. In 1-3 sentences explain whether or not the absence of type checking in Python is a good idea.

Staff opinion *The lack of type checking in Python is a bad idea. Type checking can catch programmer mistakes that would otherwise lead to frustrating run-time bugs. For example: code could silently computate incorrect answers because of* `int-float` *problems; or the program could fail after running for a long time, because a variable had an unexpected type, something that can usually be detected with type-checking; etc.*

Alternative views *We're not (particularly) dogmatic, so we accepted reasonable arguments supporting other views. For example:*

- *The lack of type checking is a good idea, because it makes writing polymorphic code easier. The same code can do different things depending on the types of its inputs (e.g., same function can reverse a string or a list).*
- *The lack of type checking is a good idea, because it allows for more concise programs. Variables can be reused, etc.*

Name: *Alyssa P. Hacker*

VI Specification [15 points]

Describe the ways in which the following specification is flawed.

```
def find (e, s):
"""uses binary search to find an index, i,
   such that s[i] == e"""
```

Answers did not need to find all the flaws to get complete credit.

Some flaws in this specification:

- *It should not limit the choice of algorithm to binary search.*
  (should: say something like 'uses an O(nlogn) search')
- *It doesn't say what the function returns, if anything.*
- *It doesn't say what happens if* `e` *is not in* `s`.
- *It doesn't specify the type of* `s`.
  (note that the type of `e` *doesn't really matter here).*

End of Quiz I

Name: *Alyssa P. Hacker*