Name:

Instructions:

1. Write your name at the top of *this* page.
2. This is a closed-book exam. No form of communication is permitted (eg, talking, texting, etc.), except with the course staff.
3. No electronic devices are permitted.
4. There are 30 multiple-choice/short-answer questions in this exam, each worth 2 points. You have 75 minutes to answer the questions.
5. Each question must be answered *with a pencil* as shown below. It will be marked as incorrect otherwise.

   Multiple-choice question: (A) (B) (C) (D) (E)

   Short answer question: 42
6. You may use the blank spaces for any scratch work.
7. Discussing the exam contents with anyone who has not taken the exam is a violation of the academic honesty code.

Problem 1. If `s` is a string object, `s.replace(u, v)` returns a string with all occurrences of `u` in `s` replaced with `v`, and `s.count(u)` returns the number of occurrences of `u` in `s`. For example, `"abba".replace("b", "a")` returns `"aaaa"`, `"abba".count("b")` returns `2`, and `"abba".count("z")` returns `0`. Consider the following functions:

```
def f(x, y, z=""):
    return x.replace(y, z)

def g(x, y="i"):
    z = 0
    for v in y:
        z += x.count(v)
    return z
```

a. What does `f("mississippi", "p", "i")` return?

(A) `"mssssssspps"`

(B) `"mississippi"`

(C) `"mpsspsspppp"`

(D) `"miiiiiiippi"`

(E) `"mississiiii"`

b. What does `g("mississippi")` return?

c. What does `len(f("mississippi", "i"))` return?

d. What does `g("mississippi", "pie")` return?

e. What does `g(f("mississippi", "i"))` return?

Problem 2. Consider the following functions:

```
def f(x):
    st = SymbolTable()
    for v in x:
        if v in st:
            st[v] += 1
        else:
            st[v] = 1
    return st

def g(x):
    return max(f(x).values())

def h(x):
    y = f(x)
    z = g(x)
    for v in y.keys():
        if y[v] == z:
            return v
    return -1
```

a. What does `f("mississippi")["p"]` return?

b. What is the value of the expression `f("mississippi")["s"] + f("mississippi")["m"]`?

c. What does `g("abracadabra")` return?

d. What does `h("abracadabra")` return?

e. What is the value of the expression `g("alakazam") * h("abracadabra")`?

Problem 3. Recall that the `Vector` data type represents a vector and supports the following API:

`Vector(a)`	constructs a new vector `v` with Cartesian coordinates taken from the list `a`
`v[i]`	returns the `i`th coordinate of `v`
`v + w`	returns the sum of `v` and `w`
`v - w`	returns the difference of `v` and `w`
`v.scale(alpha)`	returns the scalar product of float `alpha` and `v`
`v.dot(w)`	returns the dot product of `v` and `w`
`v.direction()`	returns the unit vector in the same direction as `v`
`abs(v)`	returns the magnitude of `v`
`len(v)`	returns the dimensions of `v`
`str(v)`	returns a string representation of `v`

Now suppose you have a 4-dimensional vector $v = (0, 3, 4, 0)$.

a. What is the syntax for creating a `Vector` object `v` that represents the vector v?

(A) `v = [0, 3, 4, 0]`

(B) `v = Vector(0, 3, 4, 0)`

(C) `v = Vector.__init__([0, 3, 4, 0])`

(D) `v = Vector([0, 3, 4, 0])`

(E) `v = Vector.__init__(0, 3, 4, 0)`

b. What is the syntax for computing the dot product of the vectors v and $3v$?

(A) `v.dot(v.scale(3))`

(B) `Vector.dot(Vector.scale(v, 3))`

(C) `v.dot(3 * v)`

(D) `v.dot(Vector.scale(v, 3))`

(E) `Vector.dot(v, 3 * v)`

c. What is the dot product of the vectors v and $3v$?

d. What does the code `abs(v)` translate to internally?

(A) `__abs__(self, v)`

(B) `v.__abs__(self)`

(C) `self.__abs__(v)`

(D) `v.__abs__()`

(E) `v.abs()`

e. What does `abs(v)` return?

f. Suppose `x` and `y` are two `Vector` objects. What does the code `x + y` translate to internally?

(A) `Vector.__add__(y, x)`

(B) `y.__add__(x)`

(C) `x.__add__(y)`

(D) `Vector.__add__(x, y)`

(E) `__add__(x, y)`

g. What does `len(v)` return?

h. What does the code `str(v)` translate to internally?

(A) `v.__str__()`

(B) `v.str()`

(C) `str(self, v)`

(D) `self.__str__(v)`

(E) `v.__str__(self)`

Problem 4. Consider sorting the following array of strings `a` using insertion sort (shown below), by making the call `sort(a)`:

B E H M P J T N C S

```
def sort(a, key=None):
    n = len(a)
    for i in range(1, n):
        for j in range(i, 0, -1):
            v, w = a[j], a[j - 1]
            if key:
                v, w = key(v), key(w)
            if v >= w:
                break
            _exchange(a, j, j - 1)
```

a. When `i = 5`, where does the corresponding item `J` get sorted (ie, what is its index) relative to the items before?

b. When `i = 7`, where does the corresponding item `N` get sorted?

c. When the sorting is complete, what is the value of `a[7]`?

Problem 5. Consider the following table, which gives the running time $T(n)$ in seconds for a program for various values of the input size n:

n	$T(n)$
1000	16
2000	32
4000	64
8000	128

a. What is the value of $T(n)$ if $n = 16000$?

b. What is the value of n if $T(n) = 1024$?

c. What is the running time classification for the program?

(A) Linear

(B) Quadratic

(C) Exponential

(D) Cubic

(E) Constant

Problem 6. Suppose that a minus sign in the input indicates pop the stack and write the return value to standard output, and any other string indicates push the string onto the stack. Further suppose that following input is processed:

```
A B C D - E F G - - H I J - K - L - M N O -
```

a. What is the *second* string in standard output?

b. What is the *fifth* string in standard output?

c. What are the contents (top to bottom) left in the stack?

(A) `M I H E C B A N`

(B) `O N M L K J I H`

(C) `H I J K L M N O`

(D) `N M I H E C B A`

(E) `A B C E H I M N`

Problem 7. Suppose that a minus sign in the input indicates dequeue the queue and write the return value to standard output, and any other string indicates enqueue the string into the queue. Further suppose that following input is processed:

```
A B C D - E F G - - H I J - K - L - M N O -
```

a. What is the *second* string in standard output?

b. What is the *fifth* string in standard output?

c. What are the contents (front to back) left in the queue?

(A) `N M I H E C B A`

(B) `O N M L K J I H`

(C) `A B C E H I M N`

(D) `H I J K L M N O`

(E) `I J K L M N O H`

Solution 1.

a. `E`

b. `4`

c. `7`

d. `6`

e. `0`

Solution 2.

a. `2`

b. `5`

c. `5`

d. `"a"`

e. `"aaaa"`

Solution 3.

a. `D`

b. `A`

c. `75`

d. `D`

e. `5.0`

f. `C`

g. `4`

h. `A`

Solution 4.

a. `3`

b. `5`

c. `"P"`

Solution 5.

a. `256`

b. `64000`

c. `A`

Solution 6.

a. `"G"`

b. `"K"`

c. `D`

Solution 7.

a. `"B"`

b. `"E"`

c. `D`