Question 1. [4 MARKS]

Beside each code fragment below, show the output that it would create. If it would generate an error say so, and give the reason why.

Part (a) [1 MARK]

```
L = ["this", "is", "fun"]
for x in L:
    x = x + "!"
print L
```

Part (b) [1 MARK]

```
s = "hellllo"
d = {}
for i in range(len(s)):
    d[s[i]] = i
print d
```

Part (c) [1 MARK]

```
L = [[10, 12, 14], [1, 2, 3, 4, 5], ["a", "b", "c"]]
print L[1][3]
```

Part (d) [1 MARK]

```
s = "what!sup?"
k = s.index("!")
print s[1:k-1] + s[k+1:]
```

Solution:

```
['this', 'is', 'fun']
{'h': 0, 'e': 1, 'l': 5, 'o': 6}
4
hasup?
```

Question 2. [6 MARKS]

Write the function below, according to its docstring. You must not use a for-loop in this question or your solution will earn zero.

```
def first_neg(L):
    '''L is a list of ints.  Return the index of the first element of L that is
    negative.  If none are negative, return -1.'''
```

Solution:

```
    i = 0
    while i < len(L) and L[i] >= 0:
        i += 1
    if i < len(L):
        return i
    else:
        return -1
```

Question 3. [6 MARKS]

Suppose we have two dictionaries whose values are `int`s. Define the **dictionary maximum** of the two dictionaries to be a new dictionary containing every key that is in both of the dictionaries. The value associated with a key is the maximum of the values for that key from d1 and d2. For example, if we have these two dictionaries:

```
d1 = {"a": 5, "d": 11, "c": -2, "j": 99}
d2 = {"d": 4, "j": 101, "z": 8}
```

their dictionary maximum is `{'d': 11, 'j': 101}`.

Write the function below, according to its docstring.

```
def dict_max(d1, d2):
    '''d1 and d2 are dicts whose values are ints.  Return a new dict that
    is the dictionary maximum of d1 and d2.'''
```

Solution:

```
    new_dict = {}
    for k in d1:
        if k in d2:
            new_dict[k] = max(d1[k], d2[k])
    return new_dict
```

Question 4. [8 MARKS]

Write the function below, according to its docstring.

```
def big_deposits(filename):
    '''str filename is the name of a file that stores deposits into a bank
    account. Each deposit is stored in a single line as an amount
    preceded by a dollar sign (for example: $1254.95). Return the number of
    deposits that exceed $1000.'''
```

Solution:

```
    deposits = open(filename, 'r')
    total = 0
    for line in deposits:
        if float(line[1:]) > 1000:
            total += 1
    return total
```