**Question 1.**     [5 MARKS]

Complete the following function according to its docstring description.

```
def longer(L, x):
    '''Given a list of strs L, return a new list that contains each
    element of L whose length is greater than int x.  For example,
    longer(['abcd', 'abc', 'ab'], 2) returns ['abcd', 'abc'].'''

    result = []
    for item in L:
        if len(item) > x:
            result.append(item)
    return result
```

CONT'D. . .

## Question 2.   [7 MARKS]

The `str` module in Python has a function called `split`. Without using Python's `split`, implement the following function according to its docstring description.

```python
def my_split(s):
    '''Given a str s, return a list of all the strings in s that are
    separated by the character '?'. For example, my_split('This?is??it')
    returns ['This', 'is', '', 'it'].'''

    result = []
    temp = ''
    for c in s:
        if c != '?':
            temp += c
        else:
            result.append(temp)
            temp = ''
    result.append(temp)
    return result
```

## Question 3.  [8 MARKS]

In the table below, trace the variable values during execution of the function call:

```
mystery(6)
```

For each blank in the table, fill in the specified variable's value **after** the corresponding line has executed. Write NR ("not reached") if that line was not executed. **The loop will iterate between 1 and 4 times; fill in only as many iterations as necessary.** On the last line, write the value returned by the function call.

**Recall that x % y gives the remainder when x is divided by y.**

```
def mystery(a):

    s = '01'

    d = ''

    while a > 0:

        c = a % 2

        d = s[c] + d

        a = a / 2

    return d
```

| | Show variable values **after** each line has executed: | | | |
|---|---|---|---|---|
| a: | 6 | | | |
| s: | '01' | | | |
| d: | '' | | | |
| | During iteration: | | | |
| | 1 | 2 | 3 | 4 |
| c: | 0 | 1 | 1 | NR |
| d: | '0' | '10' | '110' | NR |
| a: | 3 | 1 | 0 | NR |
| value returned: | '110' | | | |