CSC 108H1 F 2008 Test 2
Duration — 35 minutes
Aids allowed: none

Student Number: |_|_|_|_|_|_|_|_|_|

Last Name: ____________________ **First Name:** ____________________

Lecture Section: L0101 **Instructor:** Gries

*Do **not** turn this page until you have received the signal to start.*
(Please fill out the identification section above, **write your name on the back of the test**, and read the instructions below.)
Good Luck!

This test consists of 3 questions on 8 pages (including this one). *When you receive the signal to start, please make sure that your copy is complete.*
Comments are not required except where indicated, although they may help us mark your answers. They may also get you part marks if you can't figure out how to write the code.
The last sheet has Python function and method descriptions that you may find useful.
If you use any space for rough work, indicate clearly what you want marked.

# 1:	______/ 5
# 2:	______/ 7
# 3:	______/ 8
TOTAL:	______/20

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'].'''
```

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.

```
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'].'''
```

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:				
d:				
During iteration:				
	1	2	3	4
c:				
d:				
a:				
value returned:				

Space for rough work:

Space for rough work:

Short Python function/method descriptions:

```
__builtins__:
  max(x, y, z, ...) -> value
    With two or more arguments, return the largest argument.
  min(x, y, z, ...) -> value
    With two or more arguments, return the smallest argument.
  len(x) -> integer
    Return the length of the list or string x.
  range([start], stop, [step]) -> list of integers
    Return a list containing the integers starting with stop and ending with
    stop - 1 with step specifying the amount to increment (or decrement).
    If start is not specified, the list starts at 0.  If step is not specified,
    the values are incremented by 1.
float:
  float(x) -> floating point number
    Convert a string or number to a floating point number, if possible.
int:
  int(x) -> integer
    Convert a string or number to an integer, if possible.  A floating point
    argument will be truncated towards zero.
str:
  str(x) -> string
    Convert an object into its string representation, if possible.
  S.find(sub) -> integer
    Return the lowest index in S where the string sub is found or -1 if sub
    does not occur in S.
  S.index(sub) -> integer
    Like find but raises an exception if sub does not occur in S.
  S.isdigit() --> boolean
    Return True if all characters in S are digits and False otherwise.
  S.replace(old, new) --> string
    Return a copy of string S with all occurrences of the string old replaced
    with the string new.
  S.split([sep]) --> list of strings
    Return a list of the words in S, using string sep as the separator and
    any whitespace string if sep is not specified.
  S.startswith(prefix) --> bool
    Return True if S starts with the str prefix, and False otherwise.
  S.strip() --> string
    Return a copy of S with leading and trailing whitespace removed.
list:
  L.append(x)
    Append x to the end of the list L.
  L.index(value) -> integer
    Returns the lowest index of value in L.
  L.insert(index, x)
    Insert x at position index.
  L.sort()
    Sorts the list in ascending order.
```

Last Name: ____________________ **First Name:** ____________________