2. a.

```python
import re
def index_lines(fname):
    infile = open(fname,'r')
    lines = infile.readlines()
    infile.close()


    in_line = {}
    for i in range(len(lines)):
        words = re.split(r'\W+', lines[i])
        for word in words:
            if word in in_line:
                in_line[word].append(i)
            else:
                in_line[word] = [i]
    return in_line
```

Test runs:

```
>>> index = index_lines('innocents.txt')
>>> index['innocent']
[783, 1956, 2485, 4008, 5389, 6253, 8389]
>>> index['the']
Squeezed text (936 lines).
>>> index['help']
[1286, 2044, 3726, 3741, 4613, 5312, 6577, 6823, 6940, 7293, 7570, 8844, 8890, 1
2225, 14011, 14431, 14924, 17784, 18787, 18994, 19080]
>>> index['Tempest']
[41]
>>> index['CHAPTER']
[7, 11, 16, 21, 28, 35, 40, 48, 53, 59, 65, 79, 84, 90, 97, 101, 108, 114, 122,
127, 133, 143, 153, 159, 166, 172, 180, 187, 197, 204, 215, 222, 229, 234, 238,
248, 254, 259, 264, 271, 277, 282, 290, 299, 306, 312, 317, 323, 329, 336, 344,
359, 368, 379, 384, 393, 401, 408, 447, 750, 890, 1049, 1305, 1515, 1723, 2144,
2334, 2528, 2784, 2961, 3309, 3647, 3893, 4288, 4443, 4758, 5039, 5525, 5755, 60
64, 6401, 6850, 7142, 7444, 8053, 8505, 8807, 8994, 9290, 9580, 10038, 10387, 10
791, 10956, 11051, 11427, 11688, 11849, 12137, 12352, 12558, 12760, 13112, 13503
, 13790, 14216, 14558, 14871, 15215, 15658, 15855, 16329, 16688, 17241, 17390, 1
7670, 18134, 18249, 18363]
>>> index['ajodf']
Traceback (most recent call last):
  File "<pyshell#6>", line 1, in <module>
    index['ajodf']
KeyError: 'ajodf'
>>> |
```

2b.

```python
def look_up(fname):
    look = index_lines(fname)

    while True:
        ent = input('Enter word: ')
        if ent == '':
            print('Bye')
            break
        try:
            print('{} occurs on lines {}'.format(ent,look[ent]))
        except KeyError:
            print('{} not found in {}'.format(ent, fname))
```

Test runs:

```
>>> look_up('innocents.txt')
Enter word: innocent
innocent occurs on lines [783, 1956, 2485, 4008, 5389, 6253, 8389]
Enter word: help
help occurs on lines [1286, 2044, 3726, 3741, 4613, 5312, 6577, 6823, 6940, 7293
, 7570, 8844, 8890, 12225, 14011, 14431, 14924, 17784, 18787, 18994, 19080]
Enter word: CHAPTER
CHAPTER occurs on lines [7, 11, 16, 21, 28, 35, 40, 48, 53, 59, 65, 79, 84, 90,
97, 101, 108, 114, 122, 127, 133, 143, 153, 159, 166, 172, 180, 187, 197, 204, 2
15, 222, 229, 234, 238, 248, 254, 259, 264, 271, 277, 282, 290, 299, 306, 312, 3
17, 323, 329, 336, 344, 359, 368, 379, 384, 393, 401, 408, 447, 750, 890, 1049,
1305, 1515, 1723, 2144, 2334, 2528, 2784, 2961, 3309, 3647, 3893, 4288, 4443, 47
58, 5039, 5525, 5755, 6064, 6401, 6850, 7142, 7444, 8053, 8505, 8807, 8994, 9290
, 9580, 10038, 10387, 10791, 10956, 11051, 11427, 11688, 11849, 12137, 12352, 12
558, 12760, 13112, 13503, 13790, 14216, 14558, 14871, 15215, 15658, 15855, 16329
, 16688, 17241, 17390, 17670, 18134, 18249, 18363]
Enter word: jdsaofhjoi
jdsaofhjoi not found in innocents.txt
Enter word: 12312
12312 not found in innocents.txt
Enter word:
Bye
>>> |
```

3.

```python
def first_digit(n):
    if n < 10:
        return n
    return first_digit(n//10)
```

Test runs:

```
>>> first_digit(0)
0
>>> first_digit(10)
1
>>> first_digit(3245801)
3
>>> first_digit(7498573250979)
7
>>> first_digit(3)
3
```

4.

```
def print_hidden(s):
    ch = s[0]
    if ch.isupper():
        print(ch)
    if len(s) == 1:
        return
    return print_hidden(s[-len(s)+1:])
```

Test runs:

```
>>> print_hidden('TesTInG')
T
T
I
G
>>> print_hidden('HelLo hoW aRe YoU')
H
L
W
R
Y
U
>>> print_hidden('nothing')
>>>
```

```
def hidden(s):
    if len(s) == 0:
        return s
    else:
        ch = s[0]
        if ch.isupper():
            return ch + hidden(s[1:])
        return hidden(s[1:])
```

5.

Test runs:

```
>>> hidden('TesTIng')
'TTI'
>>> hidden('heLlO')
'LO'
>>> hidden('HoW Are YoU tOdaY')
'HWAYUOY'
>>>
```