

NAME:  
EMAIL:  
SIGNATURE:  
CIRCLE COURSE SECTION: MW 11-1 TTh 1-3 TTh 4-6 TTh 6-8

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
Total	

Lehman College, CUNY  
CIS 166 & CMP 230 Exam 2, Version 1, Fall 2012

1. What will the following code print:

```
storms = "AlbertoXBerylXChrisXDebbyXErnestoXFlorenceXGordonXHelene"  
print(storms[0], storms[1], storms[-1])  
num = storms.count("X") + 1  
names = storms.split("X")  
print("The first", num, "storms are", names)  
message = names[-1].lower()  
print(message[0:3], "p!!!")
```

Output:

2. Write a **function** that takes as input a string of words separated by commas and returns the list of the words in upper case.

3. What will the following code print:

```
(a) for i in [0,2,4,8,9]:  
    j = (i+5)%10  
    print(i,j)
```

```
(b) s = "windy"  
for i in s:  
    n = ord('a')+(ord(i)-ord('a')+3)%26  
    print(i,chr(n))
```

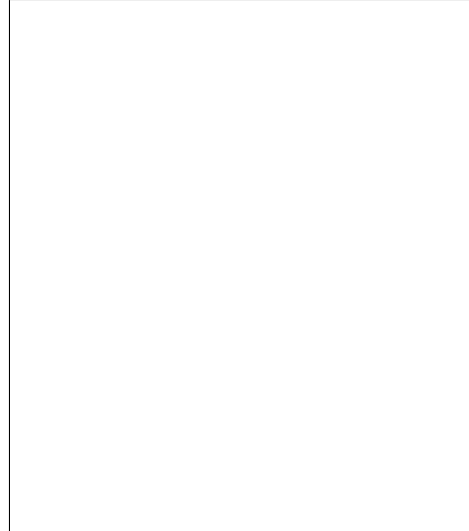
Output:

Output:

4. What will the following program print:

```
def first():
    print()
    print("And when ", end="")
def repeat(dir):
    print("they were",dir, end=" ")
    return(1)
def verse(dir):
    first()
    return (repeat(dir)+ repeat(dir))
def end():
    first()
    print("they were only half-way up,")
    print("They were neither up nor down")
    print()
def mainVersion1():
    count = verse("up")
    count = count + verse("down")
    end()
    print("Number of repeats is", count)
mainVersion1()
```

**Output:**



5. Fill in the missing function definitions for this program:

```
def main():
    welcome()           #Prints "Welcome" to the screen
    x,y,z = userInput() #Asks user for 3 inputs and returns numbers entered
    d = calculate(x,y,z) #Returns the sum of the parameters
    displayResults(d)  #Prints the result to the screen
main()
```

(That is, write the functions `welcome()`, `userInput()`, `calculate()` and `displayResults()`.)

6. What is returned when the function is invoked on the inputs below:

```
def mystery(s1):  
    n = s1.count(" ")  
    total = 1  
    for i in range(1,n):  
        total = total*i  
    return(total)
```

(a) `mystery("what does this do?")`

**Return:**

(b) `mystery("mystery, mystery?")`

**Return:**

(c) `mystery("I know, I know, I know!")`

**Return:**

7. What will the following code print:

```
nums = [1,2,10,20,2,1]  
m = 0  
M = 0  
for i in range(0,len(nums),2):  
    m = m + nums[i]  
    M = M + nums[i+1]  
print(m,M)
```

**Output:**

8. Write a function that takes as a parameter a list of strings and returns a list containing the first letter of each of the strings. That is, if the input parameter is ["This", "is", "an", "Example"], your function should return ["T", "i", "a", "E"].

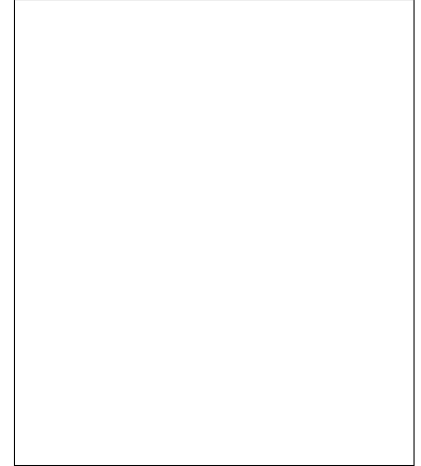
9. Given the following program and input file, what is printed:

```
def main():
    infile=open("in.txt","r")
    for line in infile.readlines():
        print(line)
        print(line.title())
    infile.close()
main()
```

**in.txt**

You must write,  
and read,  
as if your life  
depended on it.

**Output:**



10. Write a **program** that reads in a text file, `infile.txt`, and writes out the contents to another file, `outfile.txt`, in all upper case.

Useful String Methods: (from p 140 of textbook)

Function	Meaning
<code>s.capitalize()</code>	Copy of <code>s</code> with only the first character capitalized.
<code>s.center(width)</code>	Copy of <code>s</code> is centered in a field of given width.
<code>s.count(sub)</code>	Count the number of occurrences of <code>sub</code> in <code>s</code> .
<code>s.find(sub)</code>	Find the first position where <code>sub</code> occurs in <code>s</code> .
<code>s.join(list)</code>	Concatenate <code>list</code> into a string using <code>s</code> as a separator.
<code>s.ljust(width)</code>	Like <code>center</code> , but <code>s</code> is left-justified.
<code>s.lower()</code>	Copy of <code>s</code> with all characters converted to lowercase.
<code>s.lstrip()</code>	Copy of <code>s</code> with leading whitespace removed.
<code>s.replace(oldsub,newsub)</code>	Replace all occurrences of <code>oldsub</code> in <code>s</code> with <code>newsub</code> .
<code>s.rfind(sub)</code>	Like <code>find</code> , but returns rightmost position.
<code>s.rjust(sub)</code>	Like <code>center</code> , but <code>s</code> is right-justified.
<code>s.rstrip()</code>	Copy of <code>s</code> with trailing whitespace removed.
<code>s.split()</code>	Split <code>s</code> into a list of substrings.
<code>s.title()</code>	Copy of <code>s</code> with first character of each word capitalized.
<code>s.upper()</code>	Copy of <code>s</code> with all characters converted to uppercase.

NAME:  
EMAIL:  
SIGNATURE:  
CIRCLE COURSE SECTION: MW 11-1 TTH 1-3 TTH 4-6 TTH 6-8

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
Total	

**Lehman College, CUNY**  
**CIS 166 & CMP 230 Exam 2, Version 2, Fall 2012**

1. What will the following code print:

```
storms = "AndreaYBarryYChantalYDeanYErinYFelixYGabrielleYHumberto"  
print(storms[0], storms[1], storms[-1])  
num = storms.count("Y") + 1  
names = storms.split("Y")  
print("The first", num, "storms are", names)  
message = names[5].upper()  
print("H", message[1:3], "P!!!")
```

**Output:**

2. Write a **function** that takes as input a string of words separated by colons and returns the list of the words in lower case.

3. What will the following code print:

```
(a) for i in [0,1,3,7,9]:  
    j = (i+5)%10  
    print(i,j)
```

```
(b) s = "wanda"  
for i in s:  
    n = ord('a')+(ord(i)-ord('a')-1)%26  
    print(i,chr(n))
```

**Output:**

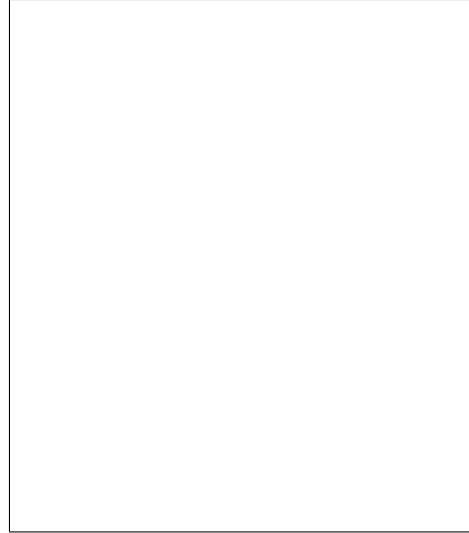
**Output:**

4. What will the following program print:

```
def first():
    print("Hickory, dickory, dock")
def repeat(dir):
    print("The mouse ran",dir)
    return(1)
def verse(n):
    print("The clock struck", n)
def mainVersion2():
    first()
    count = repeat("up the clock")
    verse(1)
    count = count + repeat("ran down")
    first()
    print()
    print("Number of repeats is", count)

mainVersion2()
```

**Output:**



5. Fill in the missing function definitions for this program:

```
def main():
    welcome()                #Prints "My program" to the screen
    x,y,z = userInput()     #Asks user for 3 inputs and returns numbers entered
    d = calculate(x,y,z)    #Returns the sum of the parameters
    displayResults(d)      #Prints the result to the screen
main()
```

(That is, write the functions `welcome()`, `userInput()`, `calculate()` and `displayResults()`.)

6. What is returned when the function is invoked on the inputs below:

```
def mystery(s1):  
    n = s1.count(" ")  
    total = 0  
    for i in range(1,n):  
        total = total+i  
    return(total)
```

(a) `mystery("what does this do?")`

**Return:**

(b) `mystery("mystery, mystery?")`

**Return:**

(c) `mystery("I know, I know, I know!")`

**Return:**

7. What will the following code print:

```
nums = [1,1,9,8,1,2]  
m = 0  
M = 0  
for i in range(0,len(nums),2):  
    m = m + nums[i]  
    M = M + nums[i+1]  
print(m,M)
```

**Output:**

8. Write a function that takes as a parameter a list of strings and returns a list containing the first letter of each of the strings. That is, if the input parameter is ["This", "is", "an", "Example"], your function should return ["T", "i", "a", "E"].



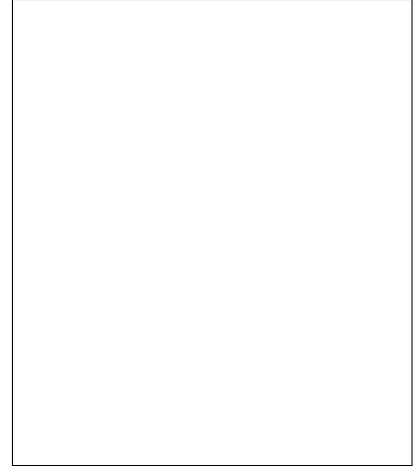
9. Given the following program and input file, what is printed:

```
def main():
    infile=open("in.txt","r")
    for line in infile.readlines():
        print(line)
        print(line.capitalize())
    infile.close()
main()
```

**in.txt**

You must write,  
and read,  
as if your life  
depended on it.

**Output:**



10. Write a **program** that reads in a text file, `infile.txt` and writes out the contents to another file, `outfile.txt`, all in lower case.

Useful String Methods: (from p 140 of textbook)

Function	Meaning
<code>s.capitalize()</code>	Copy of <code>s</code> with only the first character capitalized.
<code>s.center(width)</code>	Copy of <code>s</code> is centered in a field of given width.
<code>s.count(sub)</code>	Count the number of occurrences of <code>sub</code> in <code>s</code> .
<code>s.find(sub)</code>	Find the first position where <code>sub</code> occurs in <code>s</code> .
<code>s.join(list)</code>	Concatenate <code>list</code> into a string using <code>s</code> as a separator.
<code>s.ljust(width)</code>	Like <code>center</code> , but <code>s</code> is left-justified.
<code>s.lower()</code>	Copy of <code>s</code> with all characters converted to lowercase.
<code>s.lstrip()</code>	Copy of <code>s</code> with leading whitespace removed.
<code>s.replace(oldsub,newsub)</code>	Replace all occurrences of <code>oldsub</code> in <code>s</code> with <code>newsub</code> .
<code>s.rfind(sub)</code>	Like <code>find</code> , but returns rightmost position.
<code>s.rjust(sub)</code>	Like <code>center</code> , but <code>s</code> is right-justified.
<code>s.rstrip()</code>	Copy of <code>s</code> with trailing whitespace removed.
<code>s.split()</code>	Split <code>s</code> into a list of substrings.
<code>s.title()</code>	Copy of <code>s</code> with first character of each word capitalized.
<code>s.upper()</code>	Copy of <code>s</code> with all characters converted to uppercase.

NAME:  
 EMAIL:  
 SIGNATURE:  
 CIRCLE COURSE SECTION: MW 11-1 TTh 1-3 TTh 4-6 TTh 6-8

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
Total	

**Lehman College, CUNY**  
**CIS 166 & CMP 230 Exam 2, Version 3, Fall 2012**

1. What will the following code print:

```
storms = "ArthurZBerthaZCristobalZDollyZEdouardZFayZGustavZHanna "
print(storms[0], storms[1], storms[-1])
num = storms.count("Z") + 1
names = storms.split("Z")
print("The first", num, "storms are", names)
message = names[3].upper()
print("HE", message[2:4], "P!!!")
```

**Output:**

2. Write a **function** that takes as input a string of words separated by semicolons and returns the list of the words in upper case.

3. What will the following code print:

```
(a) for i in [2,4,6,8,9]:
    j = (i+4)%10
    print(i,j)
```

```
(b) s = "wilma"
for i in s:
    n = ord('a')+(ord(i)-ord('a')-1)%26
    print(i,chr(n))
```

**Output:**

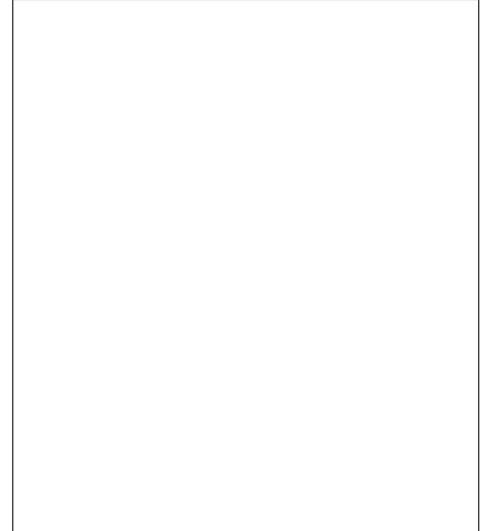
**Output:**

4. What will the following program print:

```
def first():
    print("Calico pie\nThe little Birds fly...")
def repeat(dir):
    print("They never came", dir)
    return(1)
def verse():
    count = repeat("back to me") + repeat("back")
    count = count + repeat("back")
    count = count + repeat("back to me!")
    return(count)
def mainVersion3():
    first()
    print("Till away they flew,--")
    count = verse()
    print()
    print("Number of repeats is", count)

mainVersion3()
```

**Output:**



5. Fill in the missing function definitions for this program:

```
def main():
    welcome()           #Prints "Welcome" to the screen
    s,t = userInput()  #Asks user for 2 STRINGS and returns them
    l = calculate(s,t) #Returns the sum of the lengths of the strings
    displayResults(l)  #Prints l to the screen
main()
```

(That is, write the functions `welcome()`, `userInput()`, `calculate()` and `displayResults()`.)

6. What is returned when the function is invoked on the inputs below:

```
def mystery(s1):  
    n = s1.count(" ")  
    total = 1  
    for i in range(1,n):  
        total = total*i  
    return(total)
```

(a) `mystery("why, o why, why?")`

**Return:**

(b) `mystery("very lost")`

**Return:**

(c) `mystery("I am, you are, we are!")`

**Return:**

7. What will the following code print:

```
nums = [2,1,8,8,1,2]  
m = 0  
M = 0  
for i in range(0,len(nums),2):  
    m = m + nums[i]  
    M = M + nums[i+1]  
print(m,M)
```

**Output:**

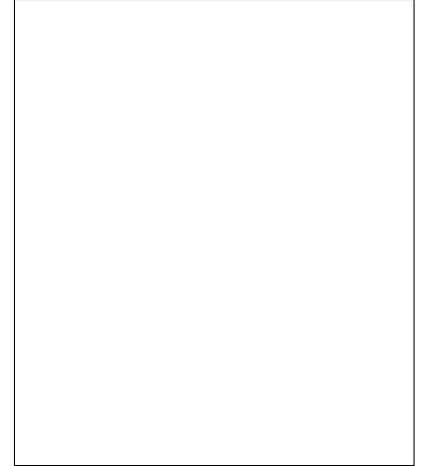
8. Write a function that takes as a parameter a list of strings and returns a list containing the length of each of the strings. That is, if the input parameter is ["This", "is", "an", "Example"], your function should return [4, 2, 2, 7].

9. Given the following program and input file, what is printed:

```
def main():
    infile=open("in.txt","r")
    for line in infile.readlines():
        print(line)
        print(line.title())
    infile.close()
main()
```

```
in.txt
Lying is done
with words
and also
with silence
```

**Output:**



10. Write a **program** that reads in a text file, `infile.txt` and writes out the contents to another file, `outfile.txt`, all in lower case.

Useful String Methods: (from p 140 of textbook)

Function	Meaning
<code>s.capitalize()</code>	Copy of <code>s</code> with only the first character capitalized.
<code>s.center(width)</code>	Copy of <code>s</code> is centered in a field of given width.
<code>s.count(sub)</code>	Count the number of occurrences of <code>sub</code> in <code>s</code> .
<code>s.find(sub)</code>	Find the first position where <code>sub</code> occurs in <code>s</code> .
<code>s.join(list)</code>	Concatenate <code>list</code> into a string using <code>s</code> as a separator.
<code>s.ljust(width)</code>	Like <code>center</code> , but <code>s</code> is left-justified.
<code>s.lower()</code>	Copy of <code>s</code> with all characters converted to lowercase.
<code>s.lstrip()</code>	Copy of <code>s</code> with leading whitespace removed.
<code>s.replace(oldsub,newsub)</code>	Replace all occurrences of <code>oldsub</code> in <code>s</code> with <code>newsub</code> .
<code>s.rfind(sub)</code>	Like <code>find</code> , but returns rightmost position.
<code>s.rjust(sub)</code>	Like <code>center</code> , but <code>s</code> is right-justified.
<code>s.rstrip()</code>	Copy of <code>s</code> with trailing whitespace removed.
<code>s.split()</code>	Split <code>s</code> into a list of substrings.
<code>s.title()</code>	Copy of <code>s</code> with first character of each word capitalized.
<code>s.upper()</code>	Copy of <code>s</code> with all characters converted to uppercase.

NAME:  
 EMAIL:  
 SIGNATURE:  
 CIRCLE COURSE SECTION: MW 11-1 TTh 1-3 TTh 4-6 TTh 6-8

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
Total	

**Lehman College, CUNY**  
**CIS 166 & CMP 230 Exam 2, Version 4, Fall 2012**

1. What will the following code print:

```
storms = "AnaWBillWClaudetteWDannyWErikaWFredWGraceWHenri"
names = storms.split("W")
print(storms[0], storms[1], storms[-1])
num = storms.count("W") + 1
names = storms.split("W")
print("The first", num, "storms are", names)
message = names[-1].lower()
print(message[0:2], "lp!!!")
```

**Output:**

2. Write a **function** that takes as input a string of words separated by dollar signs and returns the list of the words in lower case.

3. What will the following code print:

```
(a) for i in [0,2,4,8,9]:
    j = (i+4)%10
    print(i,j)
```

```
(b) s = "wendy"
for i in s:
    n = ord('a')+(ord(i)-ord('a')+3)%26
    print(i,chr(n))
```

**Output:**

**Output:**

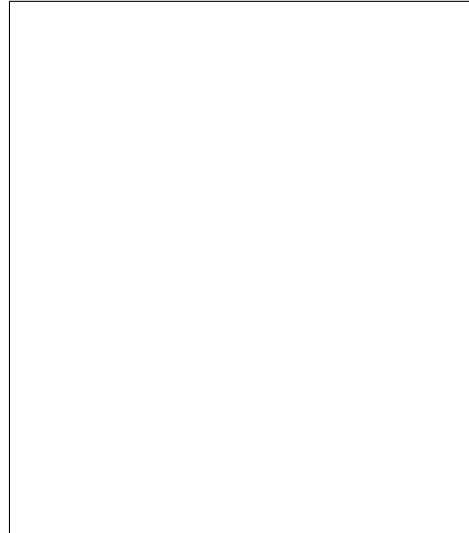


4. What will the following program print:

```
def repeat(dir):
    print("Jack", dir)
    return(1)
def verse1():
    count = repeat("be Nimble")
    count = count + repeat("be quick")
    count = count + repeat("jump over")
    return(count)
def verse2():
    count = repeat("jumped high")
    count = count + repeat("jumped low")
    count = count + repeat("jumped over")
    return(count)
def mainVersion4():
    count = verse1()
    print("The candlestick")
    count = count + verse2()
    print("and burned his toe")
    print()
    print("Number of repeats is", count)

mainVersion4()
```

**Output:**



5. Fill in the missing function definitions for this program:

```
def main():
    welcome()           #Prints "My Program" to the screen
    s,t = userInput()   #Asks user for 2 STRINGS and returns them
    l = calculate(s,t)  #Returns the sum of the lengths of the strings
    displayResults(l)  #Prints l to the screen
main()
```

(That is, write the functions `welcome()`, `userInput()`, `calculate()` and `displayResults()`.)

6. What is returned when the function is invoked on the inputs below:

```
def mystery(s1):  
    n = s1.count(" ")  
    total = 0  
    for i in range(1,n):  
        total = total+i  
    return(total)
```

(a) `mystery("why, o why, why?")`

**Return:**

(b) `mystery("very lost")`

**Return:**

(c) `mystery("I am, you are, we are!")`

**Return:**

7. What will the following code print:

```
nums = [2,1,20,10,2,1]  
m = 0  
M = 0  
for i in range(0,len(nums),2):  
    m = m + nums[i]  
    M = M + nums[i+1]  
print(m,M)
```

**Output:**

8. Write a function that takes as a parameter a list of strings and returns a list containing the length of each of the strings. That is, if the input parameter is ["This", "is", "an", "Example"], your function should return [4, 2, 2, 7].

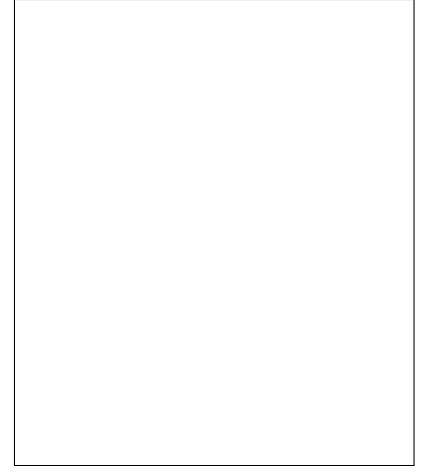
9. Given the following program and input file, what is printed:

```
def main():
    infile=open("in.txt","r")
    for line in infile.readlines():
        print(line)
        print(line.capitalize())
    infile.close()
main()
```

**in.txt**

Lying is done  
with words  
and also  
with silence

**Output:**



10. Write a **program** that reads in a text file, `infile.txt`, and writes out the contents to another file, `outfile.txt`, in all upper case.

Useful String Methods: (from p 140 of textbook)

Function	Meaning
<code>s.capitalize()</code>	Copy of <code>s</code> with only the first character capitalized.
<code>s.center(width)</code>	Copy of <code>s</code> is centered in a field of given width.
<code>s.count(sub)</code>	Count the number of occurrences of <code>sub</code> in <code>s</code> .
<code>s.find(sub)</code>	Find the first position where <code>sub</code> occurs in <code>s</code> .
<code>s.join(list)</code>	Concatenate <code>list</code> into a string using <code>s</code> as a separator.
<code>s.ljust(width)</code>	Like <code>center</code> , but <code>s</code> is left-justified.
<code>s.lower()</code>	Copy of <code>s</code> with all characters converted to lowercase.
<code>s.lstrip()</code>	Copy of <code>s</code> with leading whitespace removed.
<code>s.replace(oldsub,newsub)</code>	Replace all occurrences of <code>oldsub</code> in <code>s</code> with <code>newsub</code> .
<code>s.rfind(sub)</code>	Like <code>find</code> , but returns rightmost position.
<code>s.rjust(sub)</code>	Like <code>center</code> , but <code>s</code> is right-justified.
<code>s.rstrip()</code>	Copy of <code>s</code> with trailing whitespace removed.
<code>s.split()</code>	Split <code>s</code> into a list of substrings.
<code>s.title()</code>	Copy of <code>s</code> with first character of each word capitalized.
<code>s.upper()</code>	Copy of <code>s</code> with all characters converted to uppercase.