Boolean Problems:

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func(x):
    return (not x) or x

func(True)
```

True
A

False
B

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func(x):
    return (not x) and (6 > 10)

func(10 > 20)
```

True
A

False
B

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func(x):
    return not not x

func(15==15)
```

True
A

False
B

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func(x, y):
    return (x or y) and (not y)

func(False, True)
```

True
A

False
B

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func(x, y):
    return (x and y) or (x and not y)

func(True, False)
```

True
A

False
B

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func(x, y):
    return (y != x) and (not x or not y)

func(True, 6 < 20)
```

True
A

False
B

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func(x, y, z):
    return (z % 2) or (y and not x)

func(True, False, 20)
```

True
A

False
B

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func( x, y, z):
    return z >= x and z >= y

func(10, -3, 42)
```

True
A

False
B

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func( x, y, z):
    return x >= y and x >= z and y >= z

func(15, 20, 25)
```

True
A

False
B

Please click the corresponding letter which best represents the **return value** of the **function call** below:

```
def func( x, y, z):
    b = y > x
    return not b or (x >= z and y == z)

func(-5, 4, 4)
```

True
A

False
B

Code-tracing:

Please type the **output** of the **function call** below:

```
def func(x):
    while (x > 2):
        print(x, end=" ")
        x = x - 1

func(5)
```

Please type the **output** of the **function call** below:

```
def func(x):
    while x == 1:
        print(x, end=" ")
        x = x + 1

func(1)
```

Please type the **output** of the **function call** below:

```
def func(x):
    while x >= 1:
        j = 1
        while j <= x:
            print("*", end= " ")
            j += 1
        x -= 1
        print()
func(2)
```

Please type the **output** of the **function call** below:

```
def func(x):
    for i in range(1, x + 1):
        for j in range(1, i + 1):
            print("*", end= " ")
        print()

func(2)
```

Please type the **output** of the **function call** below:

```
def func(x):
    i = 0
    while i <= 5:
        if i == x: return
        print("Hello", end=" ")
        i += 3

func(4)
```

Please type the **output** of the **function call** below:

```
def func():
    for i in range(-1, -5, -1):
        print(i, end=" ")

func()
```

Please type the **output** of the **function call** below:

```
def func(x, y):
    z = x
    x = y
    y = z
    print(x, y)

func(10, 20)
```

Please type the **output** of the **function call** below:

```
def func(x, y):
    z = x
    x = y
    y = z
    print(x, y)

func(10, 20)
```

Please type the **output** of the **function call** below:

```
def func():
    c = 1;
    for i in range (1, 2):
        c *= i
        print(c)

func()
```

Please type the **output** of the **function call** below:

```
def func(nums):
    x = 2;
    for i in range(len(nums)):
        x += nums[i]
    print(x)

nums = [1, 2, 3, 4]
func(nums)
```

Please type the **output** of the **function call** below:

```
def func(nums):
    x = 0;
    for i in range(len(nums)):
        if nums[i] == -1: x += 1
        print(x)

nums = [-1, 2, -1]
func(nums)
```

Code-writing:

Clunker Motors Inc. is recalling all vehicles in its Extravagant line from model years 1999-2002. Given an int variable <tt>modelYear</tt> and a string <tt>modelName</tt>, print "RECALL" if modelYear and <tt>modelName</tt> match the recall details.

Write a statement that toggles the value of the bool variable <tt>onOffSwitch</tt>. That is, if <tt>onOffSwitch</tt> is false, its value is changed to true; if <tt>onOffSwitch</tt> is true, its value is changed to false

Write the definition of a function <tt>isPositive</tt>, that receives an integer parameter and returns true if the parameter is positive, and false otherwise.

Implement a function <tt>is_sorted</tt> that accepts a list of integer values and returns true if it is non-decreasing, and false otherwise.

Given a string, <tt>s</tt>, and an integer, <tt>x</tt>, update <tt>s</tt> such that its characters are shifted by <tt>x</tt>. Ex. <tt>s = "cd"</tt>, <tt>x = 1</tt>. Loop results in <tt>s = "de"</tt>.

Given integer variables <tt>amount</tt> and <tt>total</tt> both initialized to 0, write a loop that reads in values for amount from standard <tt>input()</tt> and adds them to <tt>total</tt> until <tt>total</tt> is greater than 100.

Implement a function <tt>containsDuplicate</tt> that accepts a list of integers and returns true if it contains a duplicate element, and false otherwise.

Implement a function <tt>secondGreatest</tt> that accepts a list of integers and returns the second greatest element in the list.

Implement a function <tt>countdown</tt> that accepts an integer <tt>n</tt> as a parameter and counts down from <tt>n</tt>, printing each number to standard output, separated by a space. After reaching <tt>0</tt>, print "liftoff!"