Question 1: Each question below contains a short program with an expression that is shaded in gray. Your task is to evaluate this expression and identify the correct answer, along with the data type of the result. **Write your answer AND data type on the answer sheet in the row corresponding to the question number.**

If the program crashes, write "error" as your answer and data type. If your answer is a string you do not need to include delimiters around your answer. If your answer includes a space you should indicate this by drawing a small "box" in your output to denote this. For example: `Hello□World□how□are□you?`

Question	Expression to Evaluate
1.1	`print( (3+1) / (3-1) )`
1.2	`print( 700 > 30 or 30 < 6 )`
1.3	`print( 3 % 2 + 7 // 2 )`

1.4	`x = 1234`
	`y = 1/2`
	`print( format(x+y, ">10,.2f") )`

1.5	`x = '3'`
	`y = '2'`
	`z = 3.2`
	`print( x + y * int(z) )`

Question 2: Trace the output of the following short programs. Each program will generate a single line of output. **Write your answer on the answer sheet in the row corresponding to the question number.**

If the program crashes, write "ERROR" as your answer. If your answer is a string you do not need to include delimiters around your answer. If your answer includes a space you should indicate this by drawing a small "box" in your output to denote this. For example: `Hello□World□how□are□you?`

Question	Program to evaluate

2.1

```
u = 10
f = 9
t = 23

if f > u:
    print('A', end='') # empty string
elif u > t:
    print('B', end='') # empty string
elif f > t:
    print('C', end='') # empty string
else:
    print('D', end='') # empty string

if u < f:
    print('E', end='') # empty string
elif f < t:
    print('F', end='') # empty string
else:
    print('G', end='') # empty string

if t > f:
    print('H', end='') # empty string
elif t > f:
    print('I', end='') # empty string
elif u > f:
    print('J', end='') # empty string
elif t > f:
    print('K', end='') # empty string
else:
    print('L', end='') # empty string

print('M')
```

2.2

```
d = 14
i = 6
t = 0

if i <= d:
    print('A', end='') # empty string
    if i != d:
        print('B', end='') # empty string
    elif t != i:
        print('C', end='') # empty string
    else:
        print('D', end='') # empty string
elif i <= d:
    print('lion', end='') # empty string
    if d > i:
        print('E', end='') # empty string
    elif d > t:
        print('F', end='') # empty string
    else:
        print('G', end='') # empty string
else:
    print('H', end='') # empty string
    if i < d:
        print('I', end='') # empty string
    elif d < t:
        print('J', end='') # empty string
    else:
        print('K', end='') # empty string

print('L')
```

Question 3: Given the variables 'foo', 'bar' and 'zoo', compute the result of the Boolean expression given the value of each variable. For example, if 'foo' has a value of True and 'bar' has a value of True, plug these values into the Boolean expression and compute the **final** result of the expression. **Write your answer on the answer sheet in the row corresponding to the question number** (but feel free to use this page as scratch paper)

Question	foo	bar	(foo and True) or (not(foo) and bar)
3.1	False	False	
3.2	False	True	
3.3	True	False	
3.4	True	True	

Question 4: Trace the output of the following short programs Each program will generate a single line of output. **Write your answer on the answer sheet in the row corresponding to the question number.**

If the program crashes, write "error" as your answer. If a loop results in an infinite loop write the word "INFINITE" as your answer. If your answer is a string you do not need to include delimiters around your answer. If your answer includes a space you should indicate this by drawing a small "box" in your output to denote this. For example: Hello□World□how□are□you?

<table>
<tr><th>Question</th><th>Program to evaluate</th></tr>
<tr><td>4.1</td><td><pre>
c = 0
r = 3
while True:
    c += 1
    if c < r:
        continue
    else:
        break
    print("A", end="")
print("B"*c)
</pre></td></tr>
<tr><td>4.2</td><td><pre>
word1 = "A"
word2 = "B"
while len(word1) < 5:
    word1 += word2
print(word1)
</pre></td></tr>
<tr><td>4.3</td><td><pre>
for i in range(5,10,2):
    print(i, end="") # empty string
</pre></td></tr>
<tr><td>4.4</td><td><pre>
for i in range(2):
    for j in range(2):
        print(i, j, sep="", end="#") # empty string
</pre></td></tr>
</table>

Question 5: Trace the output of the following short programs. Each program will generate a single line of output. **Write your answer on the answer sheet in the row corresponding to the question number.**

If the program crashes, write "error" as your answer. If your answer is a string you do not need to include delimiters around your answer. If your answer includes a space you should indicate this by drawing a small "box" in your output to denote this. For example: `Hello□World□how□are□you?`

Question	Program to evaluate
5.1	

```
def fun1():
    print('A', end='') # empty string

def fun2():
    print('B', end='') # empty string

def fun3():
    print('C', end='') # empty string

def fun4():
    print('D', end='') # empty string
    fun3()

def fun5():
    print('E', end='') # empty string
    fun4()

print("Q", end="") # empty string
fun5()
print("R")
```

5.2

```
def fun1(g):
    g += 3
    print(g, end=‘#’)

def fun2(p):
    p += 5
    print(p, end='#')
    fun3(p)
    return p

def fun3(w):
    w += 2
    print(w, end='#')
    return w

y = 2
print(y, end='#')
y = fun2(y)
print(y, end='#')
```

Question 6: **Write your program using the page provided on the answer sheet for this question.**

For this program you will be writing a program that generates a series of numbers within a given range (all on the same line). It should also report how many numbers were printed during the program. Here's a sample running of the program (user input is shaded):

```
Enter a starting value: 1
Enter an ending value: 10

1 2 3 4 5 6 7 8 9 10
10 numbers were printed
```

Your program should prompt the user to enter a starting value and an ending value. The program should validate the user input such that the ending value is always larger than the starting value. Here is an example that shows this behavior

```
Enter a starting value: 10
Enter an ending value: 1
Invalid ending value, try again

Enter an ending value: 2
Invalid ending value, try again

Enter an ending value: 15

10 11 12 13 14 15
6 numbers were printed
```