CSIT110 / CSIT810
Python

Lecture 3

Dr. Joseph Tonien

School of Computing and Information Technology
University of Wollongong

Objectives

Understanding of:

- Multi-line code statement
- Escape sequence
- String format
- Numerical operations

Multi-line code statement

Let’s look at the following example:

```
subject1_code = "CSCI111"
subject1_mark = 80

subject2_code = "MATH103"
subject2_mark = 75

subject3_code = "PHYS101"
subject3_mark = 85

print("Exam result. " + subject1_code + ": " +
str(subject1_mark) + ", " + subject2_code + ": " +
str(subject2_mark) + ", " + subject3_code + ": " +
str(subject3_mark) + ".")
```

this is a long line of code

What do you think this program will do?

Let’s look at the following example:

```
subject1_code = "CSCI111"
subject1_mark = 80

subject2_code = "MATH103"
subject2_mark = 75

subject3_code = "PHYS101"
subject3_mark = 85

print("Exam result. " \
  + subject1_code + ": " + str(subject1_mark) \
  + ", " \
  + subject2_code + ": " + str(subject2_mark) \
  + ", " \
  + subject3_code + ": " + str(subject3_mark) \
  + ".")
```

we can break a long line of code into multi-line

Let’s look at the following example:

```
subject1_code = "CSCI111"
subject1_mark = 80

subject2_code = "MATH103"
subject2_mark = 75

subject3_code = "PHYS101"
subject3_mark = 85

print("Exam result. " \
  + subject1_code + ": " + str(subject1_mark) \
  + ", " \
  + subject2_code + ": " + str(subject2_mark) \
  + ", " \
  + subject3_code + ": " + str(subject3_mark) \
  + ".")
```

we can break a long line of code into multi-line

Escape sequence

```
print("Welcome to Unimovies!")
print("Thursday July 30 at 7.15pm: Inside Out")
```

Program output:

```
Welcome to Unimovies!
Thursday July 30 at 7.15pm: Inside Out
```

```
print("Welcome to Unimovies!")
print("Thursday July 30 at 7.15pm: Inside Out")
```

How do we write program for this output:

```
Welcome to Unimovies!
Thursday July 30 at 7.15pm: "Inside Out"
```

How about this program?

```
print("Welcome to Unimovies!")
print("Thursday July 30 at 7.15pm: "Inside Out"")
```

what is wrong with this code?

We want to write a program for this output:

```
Welcome to Unimovies!
Thursday July 30 at 7.15pm: "Inside Out"
```

The correct program

```
print("Welcome to Unimovies!")
print("Thursday July 30 at 7.15pm: \"Inside Out\"")
```

using escape sequence

Program output:

```
Welcome to Unimovies!
Thursday July 30 at 7.15pm: "Inside Out"
```

Escape Sequence	**Meaning**
\\	Backslash (\)
\'	Single quote (')
\"	Double quote (")
\b	Backspace
\n	New line
\t	Tab

Escape sequence

```
print("Your details:\n")
print("\tName: \"Joseph Tonien\"")
print("\tSN:   \"2012345\"")
print("\nEnrolment record:\n")
print("\tMATH101")
print("\tCSCI201")
```

Program output:

```
Your details:

    Name: "Joseph Tonien"
    SN:   "2012345"

Enrolment record:

    MATH101
    CSCI201
```

Escape sequence

```
print("Escape sequence:")
print("\\n : Insert a newline.")
print("\\t : Insert a tab.")
print("\\\" : Insert a double quote character.")
print("\\\' : Insert a single quote character.")
print("\\\\ : Insert a backslash character.")
```

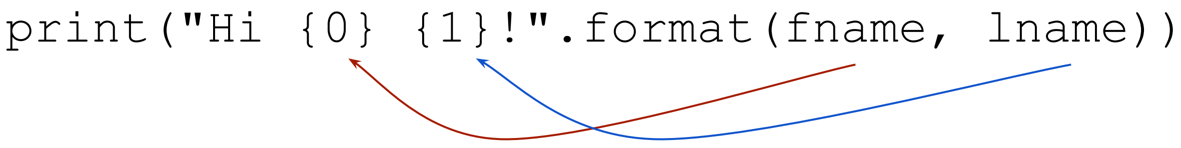
What is the output of this program?

String format

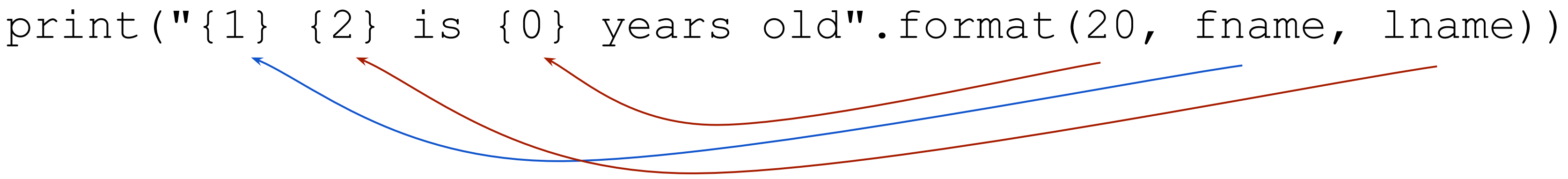
```
fname = "John"
lname = "Smith"
print("Hi {0} {1}!".format(fname, lname))
print("{1} {2} is {0} years old".format(20, fname, lname))
print("And his favorite number is {0}".format(7))
```

What is the output of this program?

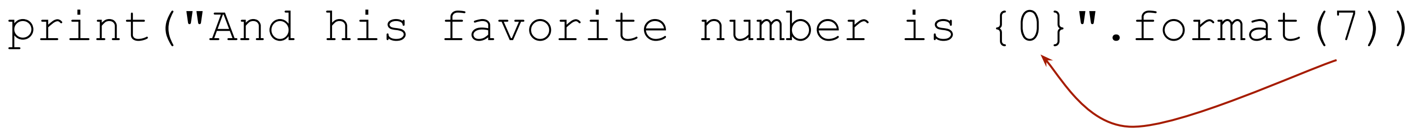
```
fname = "John"
lname = "Smith"
print("Hi {0} {1}!".format(fname, lname))
print("{1} {2} is {0} years old".format(20, fname, lname))
print("And his favorite number is {0}".format(7))
```



```
print("Hi {0} {1}!".format(fname, lname))
```



```
print("{1} {2} is {0} years old".format(20, fname, lname))
```



```
print("And his favorite number is {0}".format(7))
```

String format with alignment

```
print("Exam result:")
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("COMM104", "Commerce I", "75", "D"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("FIN201", "Accounting", "85", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("MTH202", "Analysis", "100", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("ECTE110", "Circuits", "90", "HD"))
print("1234567890123456789012345678901234567890")
```

What is the output of this program?

```
print("Exam result:")
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("COMM104", "Commerce I", "75", "D"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("FIN201", "Accounting", "85", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("MTH202", "Analysis", "100", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("ECTE110", "Circuits", "90", "HD"))
print("1234567890123456789012345678901234567890")
```

Program output

```
Exam result:
COMM104   Commerce I        75    D
FIN201    Accounting        85   HD
MTH202    Analysis         100   HD
ECTE110   Circuits          90   HD
1234567890123456789012345678901234567890
```

```
print("Exam result:")
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("COMM104", "Commerce I", "75", "D"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("FIN201", "Accounting", "85", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("MTH202", "Analysis", "100", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("ECTE110", "Circuits", "90", "HD"))
print("1234567890123456789012345678901234567890")
```

left alignment, using 10 spaces

```
Exam result:
COMM104   Commerce I        75    D
FIN201    Accounting        85   HD
MTH202    Analysis         100   HD
ECTE110   Circuits          90   HD
1234567890123456789012345678901234567890
```

```
print("Exam result:")
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("COMM104", "Commerce I", "75", "D"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("FIN201", "Accounting", "85", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("MTH202", "Analysis", "100", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("ECTE110", "Circuits", "90", "HD"))
print("1234567890123456789012345678901234567890")
```

left alignment, using 15 spaces

```
Exam result:
COMM104   Commerce I        75    D
FIN201    Accounting        85   HD
MTH202    Analysis         100   HD
ECTE110   Circuits          90   HD
1234567890123456789012345678901234567890
```

```
print("Exam result:")
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("COMM104", "Commerce I", "75", "D"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("FIN201", "Accounting", "85", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("MTH202", "Analysis", "100", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("ECTE110", "Circuits", "90", "HD"))
print("1234567890123456789012345678901234567890")
```

right alignment, using 5 spaces

```
Exam result:
COMM104   Commerce I        75    D
FIN201    Accounting        85   HD
MTH202    Analysis         100   HD
ECTE110   Circuits          90   HD
1234567890123456789012345678901234567890
```

```
print("Exam result:")
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("COMM104", "Commerce I", "75", "D"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("FIN201", "Accounting", "85", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("MTH202", "Analysis", "100", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("ECTE110", "Circuits", "90", "HD"))
print("1234567890123456789012345678901234567890")
```

right alignment, using 5 spaces

```
Exam result:
COMM104   Commerce I        75    D
FIN201    Accounting        85   HD
MTH202    Analysis         100   HD
ECTE110   Circuits          90   HD
1234567890123456789012345678901234567890
```

```
print("Exam result:")
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("COMM104", "Commerce I", "75", "D"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("FIN201", "Accounting", "85", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("MTH202", "Analysis", "100", "HD"))
print("{0:<10}{1:<15}{2:>5}{3:>5}".format("ECTE110", "Circuits", "90", "HD"))
print("12345678901234567890123456789012345678 90")
```

<	left
>	right
^	center

```
Exam result:
COMM104   Commerce I        75    D
FIN201    Accounting        85   HD
MTH202    Analysis         100   HD
ECTE110   Circuits          90   HD
1234567890123456789012345678901234567890
```

```
print("Exam result:")
print("{0:<10}{1:^15}{2:>5}{3:>5}".format("COMM104", "Commerce I", "75", "D"))
print("{0:<10}{1:^15}{2:>5}{3:>5}".format("FIN201", "Accounting", "85", "HD"))
print("{0:<10}{1:^15}{2:>5}{3:>5}".format("MTH202", "Analysis", "100", "HD"))
print("{0:<10}{1:^15}{2:>5}{3:>5}".format("ECTE110", "Circuits", "90", "HD"))
print("1234567890123456789012345678901234567890")
```

<	left
>	right
^	center

center alignment, using 15 spaces

```
Exam result:
COMM104     Commerce I      75    D
FIN201      Accounting      85   HD
MTH202       Analysis      100   HD
ECTE110      Circuits       90   HD
1234567890123456789012345678901234567890
```

```
print("{0} x {1} = {2}".format(1, 5, 1*5))
print("{0} x {1} = {2}".format(2, 5, 2*5))
print("{0} x {1} = {2}".format(3, 5, 3*5))
print("{0} x {1} = {2}".format(4, 5, 4*5))
print("{0} x {1} = {2}".format(5, 5, 5*5))
print("{0} x {1} = {2}".format(6, 5, 6*5))
print("{0} x {1} = {2}".format(7, 5, 7*5))
print("{0} x {1} = {2}".format(8, 5, 8*5))
print("{0} x {1} = {2}".format(9, 5, 9*5))
print("{0} x {1} = {2}".format(10, 5, 10*5))
```

What is the output of this program?

```
print("{0} x {1} = {2}".format(1, 5, 1*5))
print("{0} x {1} = {2}".format(2, 5, 2*5))
print("{0} x {1} = {2}".format(3, 5, 3*5))
print("{0} x {1} = {2}".format(4, 5, 4*5))
print("{0} x {1} = {2}".format(5, 5, 5*5))
print("{0} x {1} = {2}".format(6, 5, 6*5))
print("{0} x {1} = {2}".format(7, 5, 7*5))
print("{0} x {1} = {2}".format(8, 5, 8*5))
print("{0} x {1} = {2}".format(9, 5, 9*5))
print("{0} x {1} = {2}".format(10, 5, 10*5))
```

```
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50
```

```
print("{0:>2} x {1:>1} = {2:>2}".format(1, 5, 1*5))
print("{0:>2} x {1:>1} = {2:>2}".format(2, 5, 2*5))
print("{0:>2} x {1:>1} = {2:>2}".format(3, 5, 3*5))
print("{0:>2} x {1:>1} = {2:>2}".format(4, 5, 4*5))
print("{0:>2} x {1:>1} = {2:>2}".format(5, 5, 5*5))
print("{0:>2} x {1:>1} = {2:>2}".format(6, 5, 6*5))
print("{0:>2} x {1:>1} = {2:>2}".format(7, 5, 7*5))
print("{0:>2} x {1:>1} = {2:>2}".format(8, 5, 8*5))
print("{0:>2} x {1:>1} = {2:>2}".format(9, 5, 9*5))
print("{0:>2} x {1:>1} = {2:>2}".format(10, 5, 10*5))
```

we want a better output

```
 1 x 5 =  5
 2 x 5 = 10
 3 x 5 = 15
 4 x 5 = 20
 5 x 5 = 25
 6 x 5 = 30
 7 x 5 = 35
 8 x 5 = 40
 9 x 5 = 45
10 x 5 = 50
```

Arithmetic operators

+	Addition	3 + 5 = 8 3 + 5.0 = 8.0 1.2 + 3.4 = 4.6
-	Subtraction	5 - 2 = 3 5 - 2.0 = 3.0 6.5 - 1.2 = 5.3
*	Multiplication	5 * 2 = 10 5 * 2.0 = 10.0 6.5 * 1.3 = 8.45

/	Division	10/2 = 5.0 10/4 = 2.5 10/2.0 = 5.0 10.0/1.2 = 8.3333
//	Floor division	10//2 = 5 10//4 = 2 10//2.0 = 5.0 10.0//1.2 = 8.0

What is the difference between Division and Floor division?

/	Division	10/2 = 5.0 10/4 = 2.5 10/2.0 = 5.0 10.0/1.2 = 8.3333
//	Floor division	10//2 = 5 10//4 = 2 10//2.0 = 5.0 10.0//1.2 = 8.0

Note that division of two integers give a decimal number

```
10/2 = 5.0
```

So if we want integer result, we should use Floor division

```
10//2 = 5
```

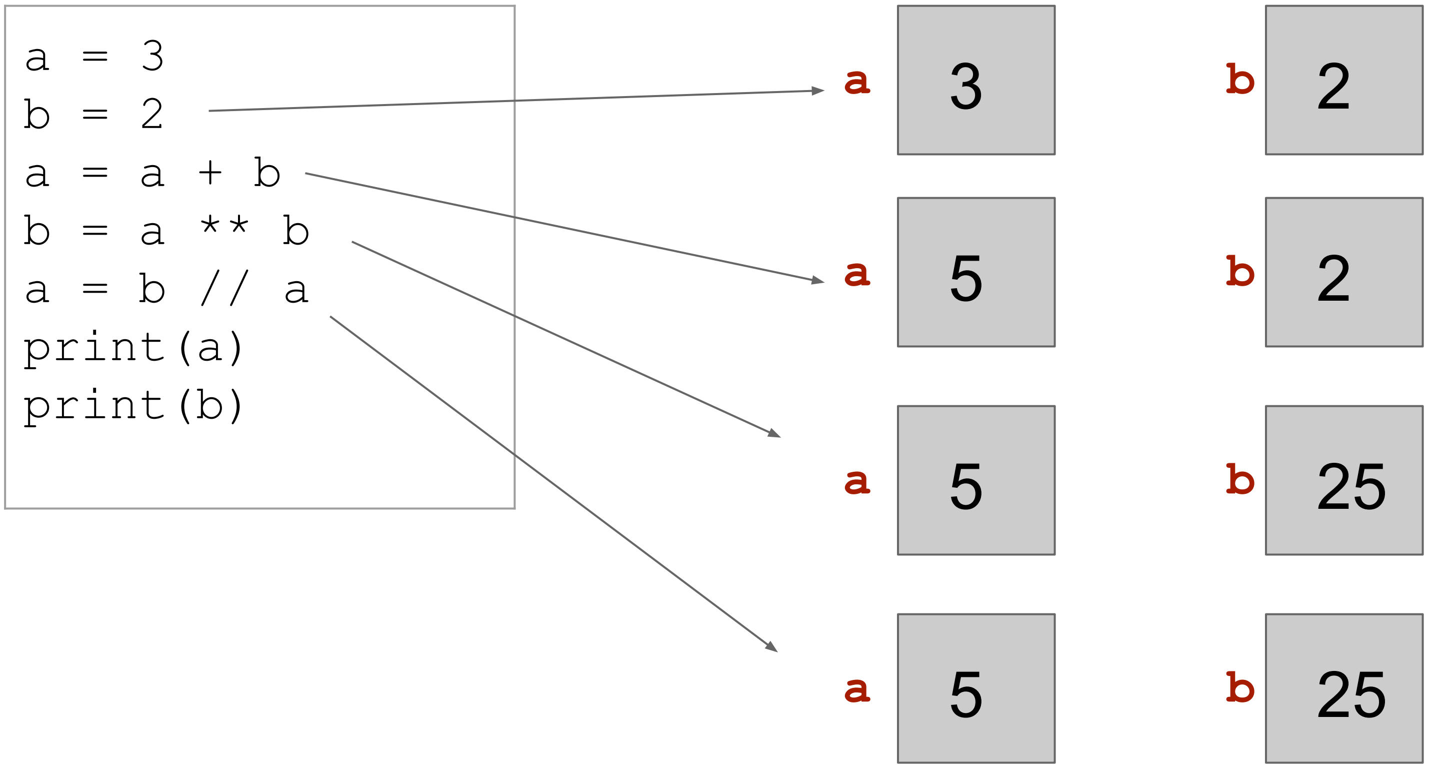
**	Exponent	10**2 = 100 10**4 = 10000 1.1**2 = 1.21 16**0.5 = 4.0 36**0.5 = 6.0

16**0.5   square root of 16

%	Modulus	15%2 = 1 124%10 = 4 28%2 = 0 37%5 = 2 -15%2 = 1

when x is an odd number: x%2 = 1
when x is an even number: x%2 = 0

to find the last digit of positive integers:
124%10 = 4
23%10 = 3



```
a = 3
b = 2
a = a + b
b = a ** b
a = b // a
print(a)
print(b)
```

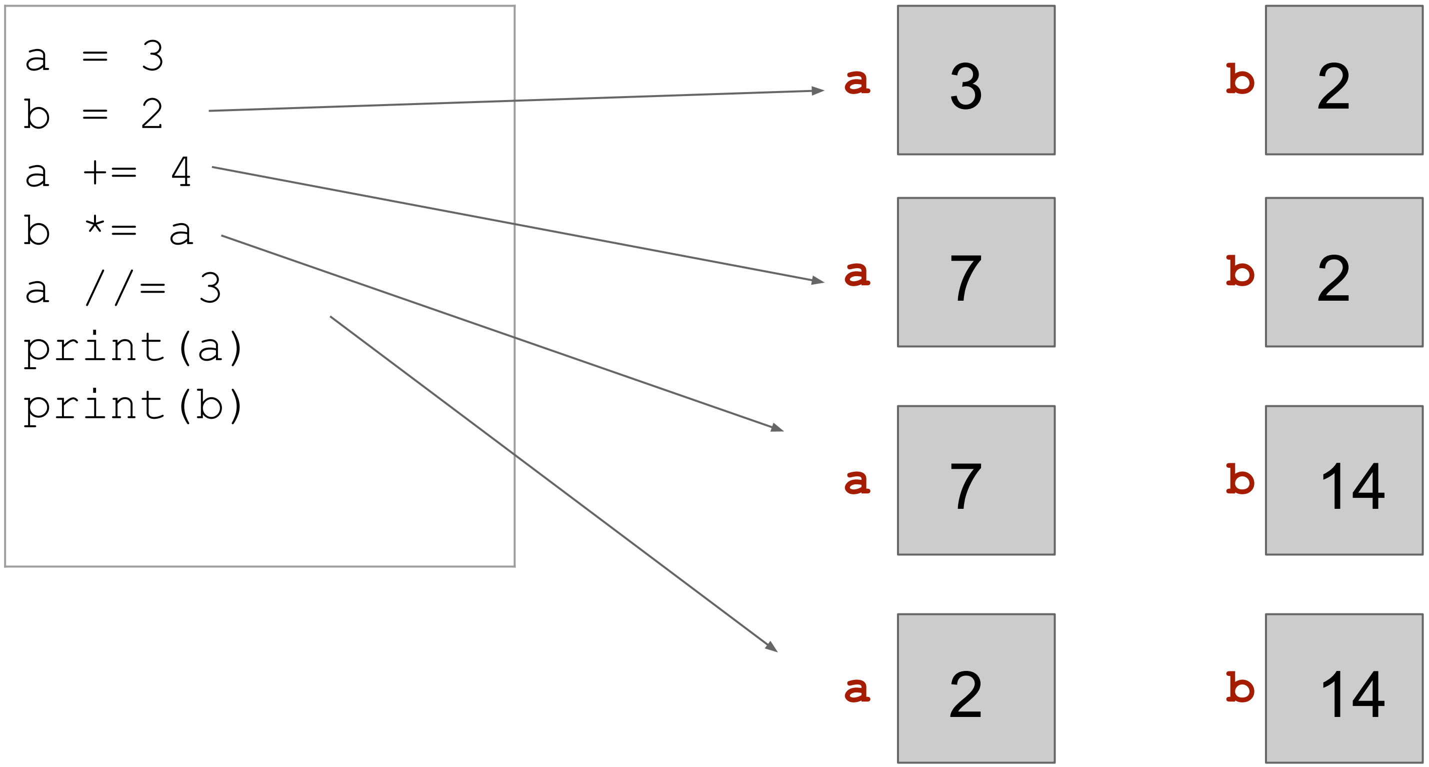
a 3 b 2

a 5 b 2

a 5 b 25

a 5 b 25

+=	x += 2 is the same as x = x + 2
-=	x -= 2 is the same as x = x - 2
*=	x *= 2 is the same as x = x * 2
/=	x /= 2 is the same as x = x / 2
//=	x //= 2 is the same as x = x / 2
**=	x **= 2 is the same as x = x ** 2
%=	x %= 2 is the same as x = x % 2

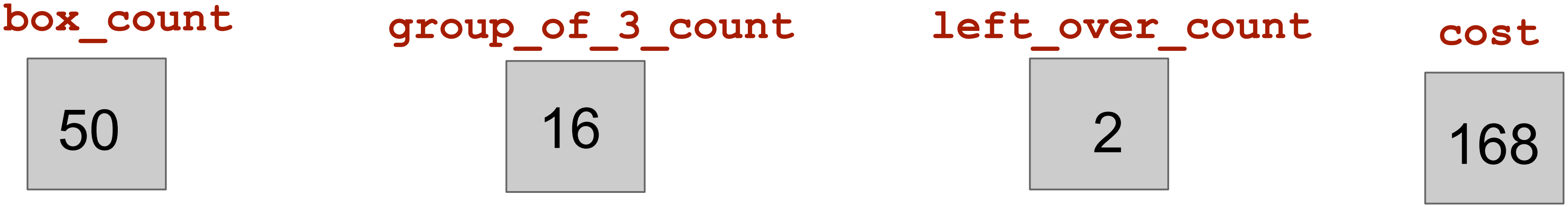


```
a = 3
b = 2
a += 4
b *= a
a //= 3
print(a)
print(b)
```

```
# Candy Box: $4/each or $10/for 3 boxes

box_count = 50

group_of_3_count = box_count // 3

left_over_count = box_count - 3 * group_of_3_count

cost = group_of_3_count * 10 + left_over_count * 4

print("{0} candy boxes cost: ${1}".format(box_count, cost))
```

What is the output of this program?

```
# Candy Box: $4/each or $10/for 3 boxes

box_count = 50

group_of_3_count = box_count // 3

left_over_count = box_count - 3 * group_of_3_count

cost = group_of_3_count * 10 + left_over_count * 4

print("{0} candy boxes cost: ${1}".format(box_count, cost))
```



box_count

50

group_of_3_count

16

left_over_count

2

cost

168

Program output

```
50 candy boxes cost $168
```