Ex24:

又是一节常规的复习；

函数可以返回多个值，你可以直接设置多个变量放在等号左端用来储存返回值（这和C不一样，C只能return一个变量）；

同样可以设置一个变量作为列表储存多个返回值（这个好特别）；

源代码：

print("Let's practice everything.")

print('You\'d need to know \'bout escape with \\ that do:') *# 转义字符*

print('\n newline and \t tabs.')

poem = """

\tThe lovely world

with logic so firmly planted

cannot discern \n the needs of love

nor comprehend passion from intuition

and requires an explanation

\n\t\twhere there is none.

""" *# 三引号原样打印*

print("--------------")

print(poem) *# 打印字符串*

print("--------------")

five = 10 - 2 + 3 - 6

print(f"This should be five: {five}") *# f-string*

def secret\_formula(*started*): *# 定义函数*

jelly\_beans = started \* 500

jars = jelly\_beans / 1000

crates = jars / 100

*return* jelly\_beans, jars, crates *# 返回变量*

start\_point = 10000

beans, jars, crates = secret\_formula(start\_point) *# 调用函数并把函数返回值赋给变量*

*# remember that this is another way to format a string*

print("With a starting point of: {}".format(start\_point)) *# .format方法，；类似C？*

*# it's just like with an f"" string"*

print(f"We'd have {beans} beans, {jars} jars, and {crates} crates.") *# f-string方法*

start\_point = start\_point / 10

print("We can also do that this way:")

formula = secret\_formula(start\_point) *# 储存至列表*

*# this is an easy way to apply a list to a format string*

print("We'd have {} beans, {} jars, and {} crates.".format(\*formula)) *# 解包列表\*列表*