[CS101] Introduction to Programming
2015 Spring - Final Examination

SECTION	STUDENT ID	NAME

※ Please check if you received all 16 pages of the test material.
※ 시작하기 전에 반드시 페이지의 수를 확인 하십시오.(전체 : 16쪽)

※ Fill in your student identification number and name. Otherwise you will lose 1 point for each missing piece of information.
※ 위의 정보(학번,이름)를 정확히 기입하지 않을 경우, 각 실수 당 1점이 감점 됩니다.

※ **TAs will not answer your questions about the exam**. If you think that there is anything ambiguous, unclear or wrong about a problem, please write the reasons and make necessary assumptions to solve the problem. We will take your explanation into consideration while grading.
※ **시험시간동안 질문을 받지 않습니다**. 만일 문제에 오류나 문제가 있을 경우, 왜 문제가 이상이 있다고 생각하는지에 대해서 기술하시면 되겠습니다. 또한 문제가 애매하다고 생각되는 경우 문제를 푸실 때 본인이 생각하는 가정을 함께 작성하셔서 문제를 푸시면 되겠습니다. 채점 시 가정 및 설명을 고려하도록 하겠습니다.

1. (20 points) Answer each question according to the instruction.

1-1 (2 points) Choose your answer to fill the blank.

Objects can have a special method _____ , called a constructor. Whenever an object of this type is created, this constructor is called.

1. __constructor__
2. __construction__
3. __init__
4. __str__
5. __ctor__

_____ (2 point)

1-2 (2 points) What does the following program output?

```
regions = ["Demacia", "Noxus", "Zaun", "Piltover", "Bilgewater", "Frejlord", "Ionia"]
print regions[1]
print regions[-5]
```

_____ (1 point)

_____ (1 point)

1-3 (3 points) What does the following program output?

```
list1 = ["A","B","C"]
list2 = list1
list2.append("D")
list1[2] = "E"
print len(list1)
print len(list2)
print list1 is list2
```

_____ (1 point)

_____ (1 point)

_____ (1 point)

1-4 (4 points) Complete the code for Fibonacci number function **fib(n)** based on the definition below.

Definition

The Fibonacci sequence or Fibonacci numbers are the numbers in the following integer seguence: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ...

By definition, the first two numbers in the Fibonacci sequence are 0 and 1, and each subsequent number is the sum of the previous two.

Code

```
def fib(n):
    if n == 0:
        return 0
    elif n == 1:
        return 1
    else :
        _____
```

Example

```
>>> print fib(5)
5
>>> print fib(10)
55
>>> print fib(20)
6765
```

_____ (4 point)

1-5 (9 points) What does the following program output? You already have the first line of the results in the answer sheet.

```
a = 5
def f(a):
    print "F = ", a

def g():
    global a
    f(a + 11)
    a = a + 17
    print "G = ", a

def h(a):
    a = 99
    g()
    print "H = ", a

print "A = ", a
f(7)
print "A = ", a
g()
print "A = ", a
h(13)
print "A = ", a
```

A = 5	(0 point)
	(1 point)
	(1 point)
	(1 point)
	(1 point)
	(1 point)
	(1 point)
	(1 point)
	(1 point)
	(1 point)

2. (20 points) Following questions are simplified versions of Homework 3.

The following program is to edit images and display the edited images. Note that it is implemented by using cs1media module instead of direct use of Image object. In addition, coordinates and pixel intensities are tuple objects of integer values. Fill in the blank to complete the program.

```
from cs1media import *

def show_img(img_fname):
    img = load_picture(img_fname)
    img.show()

def blur(img_fname):
    img = load_picture(img_fname)
    width, height = img.size()
    new_img = create_picture(width, height)

    for x in range(width):
        for y in range(height):
            if not ( #2-1 (Condition to ignore borderline pixels) ):
                R = G = B = 0
                for i in range(-1, 2):
                    for j in range(-1, 2):
                        pixel = img.get(x+i, y+j)
                        # 2-2
                            (Calculate average of 9 surrounding  pixels)
                # 2-3 (Update pixel information of edited image)
    new_img.show()

def merge(img_fname1, img_fname2):
    img1 = load_picture(img_fname1)
    img2 = load_picture(img_fname2)
    width_img1, height_img1 = img1.size()
    width_img2, height_img2 = img2.size()

    width_new, height_new = 0, 0
```

```
    # 2-4 (Set width and height of edited image)
    new_img = create_picture(width_new, height_new)

    for x in range(width_new):
        for y in range(height_new):
            R1, R2, G1, G2, B1, B2 = 0, 0, 0, 0, 0, 0
            if x < width_img1 and y < height_img1:
                R1, G1, B1 = img1.get(x, y)
            if x < width_img2 and y < height_img2:
                R2, G2, B2 = img2.get(x, y)
            new_img.set(x, y, ( (R1+R2) / 2, (G1+G2) / 2, (B1 + B2) / 2 ) )
    new_img.show()

def main():
    while True:
        process = raw_input("Quit(q), Show(s), Blur(bl), or Merge(m)?: ")
        if process == 'q':
            return
        elif process == 's':
            fname = raw_input("please type the file name")
            show_img(fname)
        elif process == 'bl':
            fname = raw_input("please type the file name")
            blur(fname)
        elif process == 'm':
            fname1 = raw_input("please type the first file name")
            fname2 = raw_input("please type the second file name")
            merge(img_fname2=fname1, img_fname1=fname2)
        else:
            print("command unknown, please refer to the menu")

main()
```

2-1 (5 points) Write down **a condition** to indicate the given position (x, y) is in borderline of the image. You should not include a 'not' keyword in the answer.

2-2 (5 points) Write down **statements** to calculate average pixel component values of surrounding 9 pixels. For example, a pixel value of position (10, 10) is determined by pixel values at positions (9,9), (9,10), (9,11), (10,9), (10,10), (10,11), (11,9), (11,10), and (11,11).

2-3 (5 points) Write down **a statement** to update pixel information of the edited image.

2-4 (5 points) Write down **statements** to set width and height of the edited image (i.e., width_new and height_new, respectively). Notice that the program can merge two images with different size. Uncovered portions by two images will be set to black.

3. (20 points) Answer each question according to the instruction.

3-1 (10 points) What is the results of the following program?

```
def Average(data, start = 0, end = None):
    if not end:
        end = len(data)
        return sum(data[start:end]) / float(end-start)

def Change(data, upper_bound):
    data = []
    for i in range(upper_bound):
        for j in range(i):
            if i%2 == 0:
                data.append(j)
            else:
                data.append(2*j)
    print data                                    (3-1-1)
    data.pop()

#main function
list_int = [2, 3, 6, 8, 11]
Change(list_int, 4)
print list_int                                    (3-1-2)
print "%d" % Average(list_int)                    (3-1-3)
print "%.2f" % Average(list_int, 2)               (3-1-4)
```

3-1-1. (3 points)

3-1-2. (2 points)

3-1-3. (2 points)

3-1-4. (3 points)

3-2 (10 points) A function 'Generate_student_ID' is implemented to generate a list of unique student ID. Fill in the blank to complete the function following the **description** and **requirements**.

	Input	Output
Description	**start**: start year (int) **end**: end year (int) **num**: the number of unique student ID number (int)	A list of unique student ID (int)

Requirements
(1) You should check if input parameters meet two conditions like followings : (1-1) '**start**' must be less than 'end' (1-2) '**num**' must be bigger than zero If (1-1) and (1-2) are not met, empty list should be returned. (2) You should specify proper input values for '**randint**' function in order to generate 8 digit numbers between ('start0000') and ('end9999'). (e.g.) if start = 2000 and end = 2010, then the range of the result values are from 20000000 to 20109999 (3) Each element in the resulting list should be unique. (4) The length of the resulting list must be same as input parameter '**num**'.

Example for 'randint' function in 'random' module
import random ex_list = [] for i in range(10) :   ex_list .append(random.randint(1, 5)) print ex_list >>> [1, 5, 4, 3, 5, 3, 1, 3, 1, 2]

[Program Code]

```
import random

def Generate_student_ID( start, end, num ) :
    ID_list = []
    if (1000<=start<10000) and (1000<=end<10000) and (3-2-1) :
        while True :
            rand_num = random.randint( (3-2-2) , (3-2-3) )
            if (3-2-4) :
                ID_list.append( rand_num )
            if (3-2-5) :
                break
    return ID_list

print Generate_student_ID( 2010, 2015, 5 )
```

[Result]

```
>>> [20144134, 20124316, 20133567, 20159866, 20158888]
```

3-2-1. (2 points)

3-2-2. (2 points)

3-2-3. (2 points)

3-2-4. (2 points)

3-2-5. (2 points)

4. (20 points) Answer each question according to the instruction.

4-1. (4 points) What is the result of the following program?

<planets.txt>

```
Mercury↲
Venus↲
Earth↲
Mars↲
Jupiter↲
Saturn↲
Uranus↲
Neptune↲
```

※ Note: "↲" denotes new line for your info (readability).
It's not visible in text editor like Notepad.

```
f = open('planets.txt')
for l in f:
  a = l.strip()
f.close()
print a
```

(4-1)	

4-2. (8 points) You are asked to make a program which count word in the text file. The program has a function named word_count and a word is only if separated by a space. (Eg, **"A total of 2,678 students (522 for Doctor's,"** has 8 words.) Your program must show the same results as the example below. And your program must be terminated when an empty string is given as a user input.

```
def word_count(filename):
   f = open(filename, 'r')
   count = 0
```

(4-2-1)

```
while True:
   filename = raw_input('Which file do you want to count word: ')
```

(4-2-2)

<kaist1.txt>

A total of 2,678 students (522 for Doctor's, 1,241 for Master's, and 915 for Bachelor's) graduated.↲
Twin brothers received their Ph.D. degrees together.↲
KAIST held its commencement ceremony on February 13, 2015. Approximately 8,000 people including the graduating class, their families, faculty, staff, and friends attended the ceremony and celebrated this milestone.↲
President Steve Kang of KAIST congratulated the graduating students and stressed the importance of their "contribution to social growth with a compassionate heart and expertise" in his commencement address. He also mentioned that all graduates would be recognized as a "Very Important Person (VIP)" and laid out the essential characteristics of what constitutes a "true VIP": vision, innovation, and perseverance.↲

※ Note: "↲" mark meaning new line just for your info (readability).
It's not visible thing in text editor like Notepad.

Example of the program

```
>>>[evaluate 4-2.py]
Which file do you want to count word: kaist1.txt
112
Which file do you want to count word:
>>>
```

(4-2-1)	

(4-2-2)	

4-3. (8 points) We crawled few recent articles in Instagram (instagram is social network service like a Facebook). These articles stored to 2 files; <tag.txt> and <url.txt>. You are asked to make two function. First, a function named **hashtag** making a file of hashtags from following <tag.txt>. The file's format like this:

<tag_1>,<tag_2>,<tag_3>,..., <tag_n>↲

...

The correct format for a hashtag is the hash symbol – the <#> immediately before a word. (Eg. '#kaist #complete #geese ...') Each hashtag delimit a space " ". Therefore, result of file would be like this:

#<tag_1> #<tag_2> #<tag_3> ... #<tag_n>↲

...

Second, a function named **filename** should make a file that contains just file names from following <url.txt>. The format of file in the <url.txt> is as in the following:

https://<path>/.../<filename>,<date>↲

...

And the result of filename function would be a series of file names, seperated by new line:

<filename>↲

...

Fill in the blank to complete the program. Two functions have no return value, they just make a file. All contents of line index at the result files must be the same as input (original) files.

```
def hashtag(tagfile, outfile):
    f = open(tagfile)
    fout = open(outfile, 'w')
    for l in f:
        (4-3-2)

def filename(urlfile, outfile):
    f = open(urlfile)
    fout = open(outfile, 'w')
    for l in f:
        (4-3-2)

hashtag('tag.txt', 'hashtag.txt')
filename('url.txt', 'filename.txt')
```

<tag.txt>

gift,britabroad,snacks,students,expat,asia,sugar,finalexam,savory,china,
sweetthings,internationalchildrensday↲
hrm,goodluck,uum,mtt,finalexam,paper,2015,sem6↲
unisza,sahabatpti,finalexam,syababmusafirilmu,imtihannihaie↲
...
finalexam,fatinizzwa,nextweek,sriangkasaapartment,stdyweek,girls↲
finalexam,theshowmustgoon↲

※ **Note: "↲" denotes <u>new line</u> just for your info (readability).**
It's not visible thing in text editor like Notepad.

<url.txt>

https://scontent.cdninstagram.com/hphotos-xaf1/t51.2885-15/e15/11282673_
988427651169366_106020714_n.jpg,2015-06-01 07:26:35↲
...
https://scontent.cdninstagram.com/hphotos-xfa1/t51.2885-15/e15/11355263_
1622636907979322_906817912_n.jpg,2015-06-01 06:07:20↲

※ **Note: "↲" denotes <u>new line</u> just for your info (readability).**
It's not visible thing in text editor like Notepad.

(4-3-1)	

(4-3-2)	

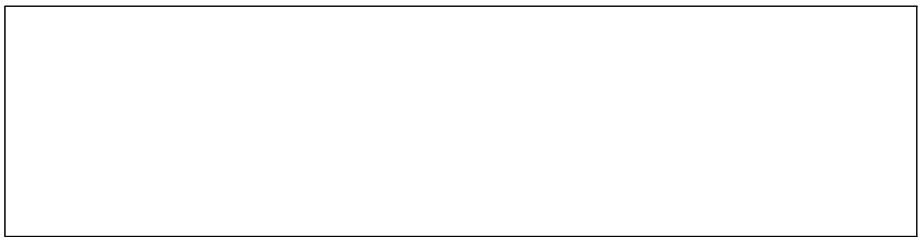
5. (20 points) Fill in the blanks to complete a class 'Point' so that the following example of interactive mode works well.

Sample code with output in interactive mode

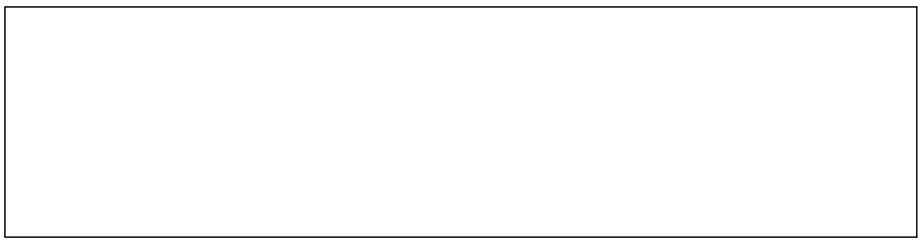
```
>>> p = Point(10, -7)
>>> print p.x, p.y
10 -7
>>> print p
(10, -7)
>>> print Point()
(0, 0)
>>> print Point(2, -3) == Point(2, -3)
True
>>> print Point(2, -3) != Point(2, -3)
False
>>> p1 = Point(-5, 5)
>>> p2 = p1.getSymmetricPoint()
>>> print p1, p2
(-5, 5) (5, -5)
>>> p2.add(Point(-100, -100))
>>> print p2
(-95, -105)
```

```
class Point (object):
    def __init__(                                    ):
        assert type(x) == int and type(y) == int
                                                     

    def __str__(self):
                                                     

    def __eq__(self, rhs) :
                                                     
```

```
def __ne__(self, rhs):
```



```
#add the given point to the current point
def add(self, p):
```



```
#return the symmetric point of the current point from (0, 0)
def getSymmetricPoint(self):
```