CSC 108H1 S 2009 Test 1
Duration — 35 minutes
Aids allowed: none

Student Number: └┴┴┴┴┴┴┴┴┘

Surname Name: ______________________ **First Name:** ______________________

*Do **not** turn this page until you have received the signal to start.*
(Please fill out the identification section above, **write your name on the back of the test**, and read the instructions below.)
Good Luck!

This midterm consists of 4 questions on 4 pages (including this one). *When you receive the signal to start, please make sure that your copy is complete.*
Comments are not required except where indicated, although they may help us mark your answers. They may also get you part marks if you can't figure out how to write the code.
If you use any space for rough work, indicate clearly what you want marked.

1: ______/ 6

2: ______/ 6

3: ______/ 8

4: ______/ 5

TOTAL: ______/25

Question 1. [6 marks]

The program below creates and displays a picture.

```
import media

def create_graphic(width, height):
   pic = media.create_picture(width, height)
   media.add_rect_filled(pic, 0, 0, width, height, media.gray)
   for pixel in pic:
      x = media.get_x(pixel)
      y = media.get_y(pixel)
      if x < width / 2 and y < height / 2:
            media.set_color(pixel, media.blue)
   return pic

if __name__ == '__main__':
   pic = create_graphic(100, 50)
   media.show(pic)
```

Draw the picture that is displayed. Label any colours clearly, and indicate all relevant dimensions.

Question 2. [6 marks]

For Assignment 1, you wrote function `average_red`:

```
def average_red(pic):
    '''Given a picture pic, return a float that is the average red value
    among all its pixels.'''
```

Complete the following function according to its docstring description. You should call `average_red` in your function. You may assume that it works correctly. You do not have to import it.

```
def extreme_red(pic):
    '''Modify Picture pic so that its red values are all extreme: Give each pixel whose
    red value is greater than the average red value of the picture a new red value of 255,
    and give all other pixels a new red value of 0.'''
```

Student #: ⎵⎵⎵⎵⎵⎵⎵⎵⎵

Question 3. [8 marks]

The left-hand column in the table below shows a series of statements to be interpreted by the Python shell. For each print statement, show the expected output in the right-hand column.

Python statements	**Output**
`print (5267 % 100) / 10`	

`list = [1,3,5,9,2,0,11]`
`print list[2:5]`	

`x = 9.6`
`y = x + 5`
`x = 20.2`
`print y`	

`name1 = "Monty"`
`name2 = "Python"`
`name1 = name2`
`name2 = name2[2:]`
`print name1`	

Question 4. [5 marks]

Suppose we have a function `valid_password` that checks whether a string can be used as a password (it must not be too short, for example):

```
def valid_password(s):
    '''Return True if string s is a valid password, and False otherwise.'''
```

The code below reads in the new password and asks the user to type it a second time. In order for the new password to be acceptable, it must be a valid password, and the user must have typed it exactly the same way twice. If not, the program will print an error message. Fill in the missing `if` condition below so that the error message will be printed if appropriate. You may call `valid_password` as needed; assume that it works correctly. You do not have to import it. Do not change or add to the code in any way other than to add the missing `if` condition.

```
    p1 = raw_input("Please enter your new password: ")

    p2 = raw_input("Please retype your new password: ")

    if

        print "Sorry, your new password has been rejected."
```

[Use the space below for rough work. This page will not be marked, unless you clearly indicate the part of your work that you want us to mark.]

Short Python function/method descriptions:

```
__builtins__:
  abs(number) -> number
    Return the absolute value of the given number.
  max(a, b, c, ...) -> value
    With two or more arguments, return the largest argument.
  min(a, b, c, ...) -> value
    With two or more arguments, return the smallest argument.
  raw_input([prompt]) -> string
    Read a string from standard input.  The trailing newline is stripped.  The prompt string,
    if given, is printed without a trailing newline before reading.
int:
  int(x) -> integer
    Convert a string or number to an integer, if possible.  A floating point argument
    will be truncated towards zero.
media:
  choose_file() --> str
    Prompt user to pick a file. Return the path to that file.
  create_picture(int, int) --> Picture
    Given a width and a height, return a Picture with that width and height.  All pixels are white.
  add_rect_filled(pic, x, y, w, h, col)
    Draw a filled rectangle of Color col, width w, and height h
    on Picture pic. The upper left corner of the rectangle is at (x, y).
  get_blue(Pixel) --> int
    Return the blue value of the given Pixel.
  get_color(Pixel) --> Color
    Return the Color object with the given Pixel's RGB values.
  get_green(Pixel) --> int
    Return the green value of the given Pixel.
  get_pixel(Picture, int, int) --> Pixel
    Given x and y coordinates, return the Pixel at (x, y) in the given Picture.
  get_red(Pixel) --> int
    Return the red value of the given Pixel.
  load_picture(str) --> Picture
    Return a Picture object from file with the given filename.
  set_blue(Pixel, int)
    Set the blue value of the given Pixel to the given int value.
  set_color(Pixel, Color)
    Set the RGB values of the given Pixel to those of the given Color.
  set_green(Pixel, int)
      Set the green value of the given Pixel to the given int value.
  set_red(Pixel, int)
    Set the red value of the given Pixel to the given int value.
  show(Picture)
    Display the given Picture.
```