

CISC106 Summer 2014 Midterm Study Guide

The midterm is composed of three main sections.

The first section will comprise multiple-choice and true/false questions. The second section will consist of reading and guessing outputs of python codes. In the third section you will be asked to write python codes.

Section-1, Section-2 and Section-3 will be worth 30,30,40 points respectively.

You are responsible from all example codes shown in class, textbook and quizzes. It's also a good practice to go over the lab questions(Especially last 3 labs).

Textbook (Starting out with Python, 3rd Edition, Tony Gaddis):

1. Chapter-1
 - a. Hardware and Software
 - b. How computers store data
 - c. Compiler, Interpreter, Assembler
 - d. Programming Languages(Low-level, High-level)
2. Chapter-2
 - a. Input, processing and output
 - b. Displaying output with print function, sep, end..etc.
 - c. Comments
 - d. Variables
 - e. Reading input from keyboard
 - f. Performing calculations
3. Chapter-3
 - a. If statement
 - b. If-else statements
 - c. Comparing strings
 - d. Nested decision structures and If-elif-else Statements
 - e. Logical operators, short-circuiting
 - f. Boolean variables.
4. Chapter-4
 - a. While and for loop.
 - b. Nested loops
 - c. Calculating a running total
5. Chapter-5: 5.1-5.8
 - a. Defining and calling a void function
 - b. Local variables
 - c. Passing arguments to functions
 - d. Global variables (general knowledge)
6. Chapter-7: 7.1-7.8
 - a. Lists

- b. Lists slicing
 - c. Finding items in the list with operators
 - d. List methods, built-in functions (append, index, insert, remove, len)
 - e. Copying a list
 - f. Two-dimensional lists
7. Chapter-8
- a. Basic string operations
 - b. String slicing
 - c. Functions for searching strings (in, not in)
 - d. String methods(isdigit, islower, isupper, lower, lstrip, rstrip, strip, upper, endswith, startswith,split)
 - e. String repetition

Example Questions

True/False Questions:

Mark the following statements as True or False.

- a) Assembler converts machine code to low-level language code
- b) Compiler converts high-level language code to machine language code
- c) Interpreter converts low-level language code to high-level language code
- d) Compiler converts high-level language code to machine language code and executes the code
- e) Data stored in computer memory is volatile.
- f) Assembly language is a high-level language
- g) Python is a high-level language
- h) For loop is a count-controlled loop
- i) While loop is a count-controlled loop
- j) One cannot write while loop inside for loop

Multiple-choice Questions:

What is not an example of an augmented assignment operator?

- a) +=
- b) -=
- c) *=
- d) <=

Code Reading Questions:

What would the following code print?

```
for i in range(2,7,2):
    if i != 4:
        print("num", i, sep="-")
    print("This is a loop")
```

Code Writing Questions:

Write a function that **gets** a list of numbers and **returns** the minimum number.

Write a function that **gets** a list of numbers and **returns** the number of odd numbers in the list.

Write 3 tests to test the following function.

```
def beATestQuestion(a):  
    if a < 5:  
        return -1  
    elif a > 5:  
        return 1  
    else  
        return 0
```