Bronx Community College
Department of Mathematics and Computer Science
Fall 2018 CSI31
Review for Midterm Exam

Chapter 1: Computers and programs

Terminology: Computer, computer science, algorithm, software, hardware, computer science
Parts of a computer system: CPU, main memory or RAM, secondary memory, input devices, output devices
Programming languages, high-level language low-level language, compiler, interpreter, syntax, semantics
Comment begins with #

Chapter 2: Writing simple programs

The software development process:

- Analyze the problem
- Determine specification
- Create a design
- Implement the design
- Test/debug the program
- Maintain the program

Input, process, output pattern

Identifier: name in a program. Python identifiers must begin with a letter or underscore, and can contain letters, digits, or underscores

Keyword: identifier with a special meaning in python, cannot be used for other purpose

Expression: a piece of program code that produces a data value
An expression is a literal, an identifier, or a combination of simpler expressions with operators.
Literal: a specific value, like a number 3 or 4.5 or a string “abcdef”

Operator: a symbol for a mathematical operation (+, -, *, **, /, //, %) or an operation on strings (+ concatenation) or others

Statements:
output statement : print(<expre>, <expr>, … <expr>), end parameter
assignment statement: <variable> = <expr>
semantics of assignment
variable as reference or sticky note
input statement : <variable> = input(<prompt>)
or <variable> = eval(input(<prompt>))
semantics: what is the difference?
simultaneous assignment:
<var1>, <var2>, …<varn> = <exp1>, <exp2>, … <expn>
semantics of simultaneous assignment
swapping values, use of simultaneous assignment for swapping values
for loop, definite loop, counted loop
range function

Chapter 3: Numeric data types

Python data types for numbers: float for numbers with fractional part, int for numbers without fractional part
mathematical operations: addition, subtraction, multiplications, division, integer division (//), remainder, exponentiation, absolute value

Math library, functions, import
Accumulator pattern
Factorial program

Computer representation of numbers in binary notation
Python uses as much memory as needed to represent an int number
Type conversion for mixed operations, explicit type conversion, rounding

Chapter 4 Objects and graphics

Object: has information or data and operations or functions or methods, data is stored in instance variables

Class: Object is an instance of some class that gives the pattern of variables and methods for the object. Object is created by calling a constructor method, whose name is the same as the class name. Example: p = Point(40, 50)
Dot notation: methods belonging to an object are called using dot operator notation objectname.methodname()

accessor methods return information about instance variables. Example t = p.getX()

mutator methods change the instance variables of an object
Example: p.move(5, -10)

Graphics module classes: GraphWin, Point, Line, Circle, Rectangle, Oval, Polygon, Entry
Know the constructors and other methods of these classes from the graphics reference 4.8, be able to write simple programs
Aliasing, clone method

Chapter 5 Sequences: strings, lists, files

Strings are sequences of characters.
Strings and lists have operators: concatenation +, repetition *, indexing [], slicing [:], length len ( )
Negative indices
A for loop can be used to iterate through the characters of a string, items of a list, lines of a file

Lists are more general than strings. strings are sequences of characters, while lists can contain items of any type, and items don’t have to be of the same type

Lists are mutable, items can be changed by assigning new values. Strings are immutable.

ACSII, Unicode for representing characters
ord: function for translating from character to Unicode; **chr:** function for translating from Unicode to character

built-in methods for strings on p.139
append method for lists

type conversion functions

newline character ‘\n’ , tab character ‘\t’

file processing: open a file for reading or writing assigns the file on the disk to a file object
read() method : returns the entire remaining file as a string
readline(): returns the next line of the file as a string
readlines(): returns a list of the remaining lines in the file

Exam structure:

Part I: paper and pencil, closed notes and book, no use of computer, multiple choice, true-false, short answer
Covering basic definitions and concepts, python syntax, semantics, basic programming techniques

Part II: open book, open notes, using computer, programming exercises, some short similar to the CodeLab exercises, one longer programming question