COURSE OUTLINE – subject to changes:

Mon 12 Jun  L-1  welcome to mcs 260 – computer literacy and Python programming
                  L-2  computer architecture – first steps with Python
Wed 14 Jun  L-3  the von Neumann machine – calculating in the Python shell
                  L-4  numbers, variables, and assignments – developing Python programs
Fri 16 Jun  L-5  operating systems – the os module – turtle graphics
                  L-6  syntax and semantics of languages – strings, lists, and tuples
Mon 19 Jun  Juneteenth holiday. No classes.
Wed 21 Jun  L-7  mass storage, files and databases – dictionaries in Python – anydbm
                  L-8  boolean algebra – flowcharts – conditional constructs in Python
Fri 23 Jun  L-9  transistors and logic gates – intrinsic operations on numbers and strings
                  L-10  flip-flops and registers – intrinsic operations on lists: queues and stacks

Project One due on Friday 23 June at 10am

Mon 26 Jun  L-11  adder circuits – loop constructs: the while and for – estimating π
                  L-12  simulation using random numbers – binary expansion with repeat until: break
Wed 28 Jun  L-13  top down design of programs – functions in Python
                  L-14  local and global variables – arguments of functions – functions using functions
Fri 30 Jun  L-15  functional programming – lambda forms – list comprehensions
                  L-16  organization of data on files – manipulating files with Python

Mon 3 Jul  L-17  data compression – format conversions – using buffers to process files
                  L-18  networking and the internet – markup languages – using the urllib module

Project Two due on Wednesday 5 July at 10am

Wed 5 Jul  L-19  review of the first 18 lectures
                  L-20  more review for the midterm exam
Fri 7 Jul  L-21  midterm exam on the first 18 lectures
                  L-22  second part of the midterm exam

Mon 10 Jul  L-23  software engineering – bottom up design of programs – modules in Python
                  L-24  the software development cycle and quality – modular design with Python
Wed 12 Jul  L-25  software licensing and open source – modules and packages
                  L-26  object-oriented programming – unified modeling language – classes in Python
Fri 14 Jul  L-27  data and functional attributes of classes – operator overloading
                  L-28  encapsulation, inheritance, polymorphism – wrapping and delegation
Mon 17 Jul  L-29  software testing, verification techniques – pre- and postconditions with assert
                  L-30  exception handling – defining, raising, and handling exceptions in Python

Project Three due on Wednesday 19 July at 10am

Wed 19 Jul  L-31  complexity and cost – complexity classes – timing Python programs
                  L-32  graphical user interfaces – using Tkinter, a GUI toolkit for Python
Fri 21 Jul  L-33  expression evaluation – graphing functions – using the Canvas widget
                  L-34  entering data with a scale – the development of animations
Mon 24 Jul  L-35  class hierarchies – visualization with inheritance
                  L-36  cellular automata – the game of life – a Tkinter application
Wed 26 Jul  L-37  processes and threads – lifecycle of a thread – multithreaded programming
                  L-38  an object-oriented implementation of a predator-prey simulation
Fri 28 Jul  L-39  network programming with sockets – the multiprocessing module
                  L-40  high level parallel programming – the multiprocessing module

Project Four due on Monday 31 July at 10am

Mon 31 Jul  L-41  recursive algorithms – tracing the execution of recursive functions
                  L-42  machine learning – neural networks
Wed 2 Aug  L-43  review of the first half of the course
                  L-44  review of the second half of the course
Fri 4 Aug  L-45  final exam
                  L-46  second part of the final exam