**Kishwaukee College Syllabus**

**CIS 111 - 3001**

**Logic and Program Design**

**3 Credit Hours, Fall 2017**

# **Course Description**

An introduction to programming designed to introduce common programming concepts to prepare for traditional programming courses. The topics to be covered include: structured programming concepts, flowcharts, pseudocode, number systems, Boolean logic, file processing, interactive input and output, an overview of assembly language, and an introduction to object-oriented programming concepts. Students will be expected to solve some problems using a programming language. Three hours lecture/discussion a week.

Prerequisite: None.

# **Meeting Time and Place**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lecture/Lab: |  | A-1374 |  |  |
| Time: | 11:00 A.M. - 12:15 P.M. | Tuesday, Thursday |
| Dates: | 8/22/17 - 12/15/17 |  |
| Withdrawal date: | 11/27/17 |  |
| Labor Day: | 9/4/17 | School closed |
| Fall break: | 10/13/17 | School closed |
| Thanksgiving: | 11/22/17 5PM - 11/25/17 | School closed |
| Midterm exam: | 10/17/17 | during class |
| Final exam: | 12/12/17 | 10:00 A.M. - 11:50 A.M. |

**Instructor Information**

|  |  |  |
| --- | --- | --- |
| Instructor: |  | David G. Klick |
| Office: | A-1342 |
| Email: | dklick@kish.edu |
| Phone: | 815/825-9337 |
| Website: | [kermit.kish.edu/~dklick/](http://kermit.kish.edu/%7Edklick/) |
| Backup website: | [klickfamily.com/david/school/](http://klickfamily.com/david/school/) |
| Desire2Learn: | <https://kish.desire2learn.com/> |
| Dept. Secretary: | 815/825-9303 (Shelley Lawson) |
| Office hours: | M 10:00 A.M. - 11:00 A.M., 1:45 P.M. - 2:30 P.M. T 1:45 P.M. - 2:30 P.M., 5:15 P.M. - 6:00 P.M. W 10:00 A.M. - 11:00 A.M., 1:45 P.M. - 2:30 P.M. R 1:45 P.M. - 2:45 P.M. other times by appointment |

**Expected Learner Outcomes**

Upon completion of this course, the student will be able to:

1. discuss and describe the software development process,
2. demonstrate the use of flowcharting and pseudocode in the development of programs,
3. compare and contrast the control structures used in structured programming,
4. create programs to solve problems using the repetition, selection, and sequence structures,
5. demonstrate the use of arrays in a program,
6. compare and contrast random and sequential file access,
7. write a program that uses sequential file access,
8. write a program that uses random file access,
9. convert numbers between binary, decimal, octal, and hexadecimal,
10. compare and contrast machine language and assembly language,
11. compare and contrast procedural and object-oriented languages,
12. demonstrate the use of Boolean logic in a program,
13. demonstrate the use of methods/functions/procedures in a program,
14. define some basic object-oriented programming terminology, and
15. demonstrate the use of object-oriented programming techniques.

# **Required Text and Materials**

1. University of Cape Town, et al. (2015) Object Oriented Programming in Python.

(Free legal download at [http://readthedocs.org/projects/python-textbok/downloads /pdf/1.0/](http://readthedocs.org/projects/python-textbok/downloads%20/pdf/1.0/)

1. Swaroop, C. H. (2013). A Byte of Python version 2.1. Lulu Press, Inc. (Free legal download at <https://www.gitbook.com/book/swaroopch/byte-of-python/details>)
2. Downey, A. B. (2015). Think Python. Sebastopol, California: O'Reilly Media. [ISBN-10: 1491939362, ISBN-13: 978-1491939369] (Free legal download at <http://greenteapress.com/thinkpython2/thinkpython2.pdf>)
3. Pilgrim, M. (2009). Dive into Python 3. New York, New York: Apress. [ISBN-10: 1430224150, ISBN-13: 978-1430224150] (Free legal download at <http://www.diveintopython3.net/>)
4. Internet access
5. Python (Free legal download at <https://www.python.org/>)
6. Python documentation (Free legal download at <https://www.python.org/doc/>)
7. Other software (such as a programmer's text editor) available free of charge over the Internet

# **Breakdown of Course Requirements**

10 assignments @ 40 points each 400 points

1 midterm exam @ 100 points 100 points

1 final exam @ 100 points 100 points

------------------------------------------------------------------

Total 600 points

# **Final Grade Determination**

A = 90 - 100% 540 points or more

B = 80 - 89.9% 480 - 539 points

C = 70 - 79.9% 420 - 479 points

D = 60 - 69.9% 360 - 419 points

F = below 60% less than 360 points

Grade reports will not be mailed out. Please check KishSOS, My Student Info, under Academic Profile, Grades, for grade reports.

# **Course Procedures**

1. Students are expected to attend class sessions on time and prepared (Note: CIS 123 class sessions are optional attendance). Students should bring whatever they need to take notes to every class.
2. Food and beverages are not permitted in the classrooms or labs. See a more detailed policy at <http://kermit.kish.edu/~dklick/foodDrinkPolicy.html>
3. Cellphones, music players, etc. must be turned off in class.
4. Students are expected to spend time outside of class completing assignments.
5. A familiarity with computers and the Windows operating system is expected.
6. Depending on the assignment, both digital and hardcopy versions of assignments may be required for submission. The procedure for submitting digital copies of assignments will be explained in class. Make sure you always keep a copy of all of your assignments. The instructor is NOT responsible for network failures, server failures, or student mistakes.
7. The instructor answers many questions via email. Due to the high volume of requests, submissions, and questions received via email, the instructor must prioritize responses. Most questions will be answered (or at least acknowledged) within 48 hours. If you do not get a response when you expect one, please keep in mind that your email may have failed to reach the instructor, or may have automatically been rejected by an email client or server. Please try to contact the instructor again and possibly use the phone or an in-person visit if email is failing.

# **Make-up Policy**

1. Assignments are to be turned in on time. Assignments which are not turned in on time will not be accepted unless individual arrangements are made in advance with the instructor. In unusual cases where late assignments are accepted, the cost of being late is ten percent of the total possible points for every portion of a day late, up to a maximum of three days late. For example, an assignment received twentyfive hours past its due date will lose twenty percent of its total possible point value (because it is two days late). Assignments which are received more than three days (seventy-two hours) late will not be accepted and are not worth any points. Exceptions may be made to this rule if the student contacts the instructor before the due date and makes special arrangements in advance with the instructor. All late acceptance decisions of this nature are left solely to the discretion of the instructor.

This rule does not apply once answers to an assignment have been distributed or posted. Assignments submitted after answers have been released are worth zero points even if the answers are posted one minute past the due date.

1. Answers to assignments may be posted online, handed out in class, or sent via email by the instructor. Once an answer to an assignment has been released, no further submissions for the assignment will be allowed. This rule supersedes all other rules about when late assignments may be accepted. In general, the instructor will try to wait at least forty-eight hours before posting or distributing solutions, but there is no guarantee, so get your assignments in on time.
2. Tests are to be taken at the day and time scheduled. Failure to take a test at the scheduled time may result in a grade of 0 for that test. In the case of an excusable absence or a genuine emergency, the instructor must be contacted as soon as possible, preferably before the scheduled test, to reschedule the makeup of that test in the Learning Skills Center on the day the student returns to campus.

# **Attendance Policy**

Class attendance is strongly encouraged. You are responsible for whatever was covered in class, whether you are there or not. If you must miss a class, it is your responsibility to contact the instructor and make arrangements for notes, handouts, or announcements that were missed. Although attendance is not counted toward the final grade, there may be coursework which is done during class time which may count toward the final grade and may not be able to be taken outside of class time.

# **Kishwaukee College Policies and Resources**

It is the responsibility of the student to be aware of Kishwaukee College Policies & Resources found on this link: [kish.edu/kcsyllabuspolicies](https://www.kish.edu/kcsyllabuspolicies)

# **Tentative Weekly Schedule**

Please note that this schedule and the topics covered are likely to change. Changes will be announced in class. If you are not able to attend class, it is your responsibility to find out what was covered. A more detailed schedule is provided on the course website. Assignment descriptions and due dates will also be posted on the course web site.

|  |  |  |  |
| --- | --- | --- | --- |
| **Week** | **Date** | **Topics** | **Reading** |
| 1 | 8/22 | syllabus, obtaining and installing Python 3 | BoP Installation  DiP3 Chap 0 (Installing Python) |
| 8/24 | fundamental programming concepts, basic flowcharting, pseudocode | OOPiP Introduction  ThinkPy Chap 1 (The way of the program) |
| 2 | 8/29 | comments, literals, variables, numbers, expressions, operators, output | BoP Basics  BoP Operators and Expressions  ThinkPy Chap 2 (Variables, ...)  OOPiP Python Basics  LangRef 6 (Expressions) |
| 8/31 | strings, input, formatting strings and output | BoP Basics  OOPiP Python Basics/Strings  ThinkPy Chap 8 (Strings)  DiP3 Strings  PyTut 3.1.2 (Strings)  PyTut 7.1 (formatting) |
| 3 | 9/5 | Boolean values, relational and logical operators, selection | BoP Control flow  OOPiP Selection control statements  ThinkPy Chap 5 (Conditionals ...)  LangRef 8 (Compound statements)  PyTut 4 (More Control Flow Tools) |
| 9/7 | repetition, flowcharting selection and repetition, input validation | BoP Control flow OOPiP Loop control statements  BoP Exceptions  ThinkPy Chap 5 (Iteration ...)  PyTut 4 (More Control Flow Tools) |
| 4 | 9/12 | functions, DocStrings | BoP Functions  ThinkPy Chap 3 (Functions)  OOPiP Functions |
| 9/14 | functions, flowcharting functions, hierarchy charts | BoP Functions  ThinkPy Chap 3 (Functions)  OOPiP Functions |
| 5 | 9/19 | modules, imports, turtle graphics | BoP Modules |
| 9/21 | recursion, recursion vs. iteration | ThinkPy Chap 6 (Fruitful functions) |
| 6 | 9/26 | lists | BoP Data Structures  ThinkPy Chap 10 (Lists)  DiP3 Lists  PyTut 3.1.3 (Lists)  OOPiP Collections/Lists  PyTut 5.1 (More on Lists) |
| 9/28 | arrays | LibRef 8.7 (array) |
| 7 | 10/3 | sets and dictionaries | BoP Data Structures  ThinkPy Chap 11 (Dictionaries)  DiP3 Dictionaries  DiP3 Sets  OOPiP Collections/Sets  OOPiP Collections/Dictionaries PyTut 5.4 (Sets)  PyTut 5.5 (Dictionaries) |
| 10/5 | tuples | BoP Data Structures  ThinkPy Chap 12 (Tuples)  DiP3 Tuples  OOPiP Collections/Tuples  PyTut 5.3 (Tuples and Sequences) |
| 8 | 10/10 | text file I/O | BoP Input and Output  ThinkPy Chap 14 (Files)  DiP3 Files |
| 10/12 | text file I/O, creating reports | BoP Input and Output  ThinkPy Chap 14 (Files)  DiP3 Files |
| 9 | 10/17 | Midterm Exam |  |
| 10/19 | control break logic for reports |  |
| 10 | 10/24 | binary file I/O | ThinkPy Chap 14 (Files)  DiP3 Files |
| 10/26 | binary file I/O, creating reports, control break logic | ThinkPy Chap 14 (Files)  DiP3 Files |
| 11 | 10/31 | searching | OOPiP Sorting, searching |
|  | 11/2 | sorting | OOPiP Sorting, searching |
| 12 | 11/7 | classes: objects | BoP Object Oriented Programming  ThinkPy Chap 15 (Classes and objects)  OOPiP Object-oriented programming |
| 11/9 | classes: objects | BoP Object Oriented Programming  ThinkPy Chap 16 (Classes and functions)  OOPiP Object-oriented programming |
| 13 | 11/14 | classes: methods | BoP Object Oriented  Programming  ThinkPy Chap 17 (Classes and methods)  OOPiP Object-oriented programming |
| 11/16 | classes: methods | BoP Object Oriented  Programming  ThinkPy Chap 17 (Classes and methods)  OOPiP Object-oriented programming |
| 14 | 11/21 | classes: inheritance | BoP Object Oriented  Programming  ThinkPy Chap 18  (Inheritance)  OOPiP Object-oriented programming |
| 11/23 | Thanksgiving - school closed 11/22 5 P.M. - 11/25 | |
| 15 | 11/28 | number systems | [Online tutorial](https://code.tutsplus.com/articles/number-systems-an-introduction-to-binary-hexadecimal-and-more--active-10848) |
| 11/30 | number systems | [Online tutorial](https://code.tutsplus.com/articles/number-systems-an-introduction-to-binary-hexadecimal-and-more--active-10848) |
| 16 | 12/5 | GUI, event-driven programming | OOPiP Intro to GUI |
| 12/7 | GUI, event-driven programming | OOPiP Intro to GUI |
| 17 | 12/12 | Final exam: 10:00 A.M. - 11:50 A.M., Rm. A-1374 | |