**CSIS 42 Python Programming**

**Spring 2019 Syllabus**

**Instructor:** Alexander Veselinov Stoykov

**Email:** [astoykov@gavilan.edu](mailto:astoykov@gavilan.edu) (Subject CSIS42)

**Phone:** (408) 852 - 2879 (Use email to contact me)

**Course Title:** Python Programming

**Units:** 4 units (option: letter grade or pass/no pass)

**Location:** Online, BU118

**Time:** 9:45am – 11:10am (Face-to-Face Section)

**Day:** Thursday

**Duration:** 01/28 – 05/24

**Office Hours:** After Class and Online

**Catalog Description:** Introduction to computing using Python. Study and create programs that perform various tasks, including text and file manipulation, internet scripting, data structures, testing, and practical problem solving with examples. Covers object - oriented programming and the Python Standard Library. Introduces students to the fundamental concepts of programming. This course has the option of a letter grade or pass/no pass.

**Online vs Regular Section:** You can sign up for the online or lecture version of the course, then attend lectures if you need them. You can do the entire class online. We will be working in a computer lab (Gilroy BU118), where you can do your homework, and get help from me in-person with your assignments. I am also available to answer your questions via email, or after class.

**Assignments**: All assignments including the class syllabus (this) will be stored at <http://ilearn.gavilan.edu.> Your grade is based upon the assignments completed. When you complete all assignments, you have completed the course. One chapter from the required textbook will be covered every week. Every week several programming assignments will be given for each chapter. All assignments are due a week after they were assigned.

**Grading:** This course has Credit/No Credit Option. You will normally get a grade in the class, but if you fill out a Credit/No Credit petition before 1/3 of the class has passed, you can take the class for credit/no credit. Petitions are available at the office or registrar. Please tell me if you do request a Credit/No Credit Option. In order to get a Credit, you need to earn at least a C grade. Be sure you are **NOT** registered as a Pass/No Pass if you want to receive a grade.

**Course Grading Method**   
You can click on  on the class web site at [ilearn.gavilan.edu](https://ilearn.gavilan.edu) to see your current grade.

This class will be graded according to the following method:

###### A = 94% - 100% | A- = 90% - 93% | B+ = 87% - 89% | B = 84% - 86 % | B - = 80% - 83%

###### C+ = 75% - 79% | C = 70% - 74% | D = 60% - 69 % | F = 0% - 59 %

**Student Responsibilities:** Whether you are taking the face-to-face section of this class or you are doing it online, you are expected to log into the class page at least once every week This is how I will measure your attendance. You are also expected to keep up with the assignments and will need to spend several hours every weekworking on this class.If you disappear and stop turning in assignments, I may drop you**.**

**Keep up with the class work:** Do not fall behind with the assignments. What you learned in week 1 will be used in Week 2 and so on. If you procrastinate, you will soon be lost. Plan tolog into the class each week read the posted material, and complete the posted assignments.

**Texts & Materials**: There is no textbook, the class web page is your textbook.

**Obtaining software**: You can obtain a free copy of Python compiler from [https://www.python.org/download/](https://www.python.org/download/%20) . We will use Python version 3.

You need to have Python on your computer if you plan to work at home. You can also do your work at the computer lab in BU118, which has Python 3 installed.

**Course Learning Outcomes:**

1. Describe the software development life-cycle.
2. Describe the principles of structured programming and be able to describe, design, implement, and test structured programs using currently accepted methodology.
3. Explain what an algorithm is and its importance in computer programming.
4. Recognize and construct common programming idioms: variables, loop, branch, subroutine, and input/output.
5. Define and demonstrate the use of the built - in data structures 'list' and 'dictionary'.
6. Apply idioms to common problems such as text manipulation, web page building, and working with large sets of numbers.
7. Design and implement a program to solve a real -world problem using the language idioms, data structures, and standard library.

**Incompletes:** I seldom give incompletes and never give them just because you have not done the work. Almost no one ever finishes an incomplete. According to admissions and records (A&R) rules: you are eligible for getting an incomplete grade “I” only if you have already completed 75% of the coursework. Only students who have been doing the class work and have extenuating circumstance may receive an Incomplete.

**Drops:** If you stop attending class, it is your responsibility to drop the class or you will get an “F”.

**Special needs:** If you have special needs such as hearing problem, visual problems, or other needs, please tell me after class and I will try to assist you.

Resource Center. ”**Necessary math skills:** If you are having trouble doing the math needed to solve the problems in the programming exercises, then you should take Math 233, Intermediate Algebra. We have noticed that one common reason students do not succeed in programming classes is the lack of math skills need to write programming algorithms. Requirement of many 4-year universities is that potential Computer Science (CS) students must take one year of calculus before becoming a CS major.

**Other classes to take:** Other classes you might take are CSIS 45 (C++), CSIS46 (Data Structures with C++), CSIS27 (Data Structures with Java), CSIS24 (Java), CSIS28 or CSIS12/12L.

**ADA Accommodation Statement:** “Students requiring special services or arrangements because of hearing, visual, or other disability should contact their instructor, counselor, or the Disability Resource Center.”

**Occupational/Vocational Statement:** “Occupational/Vocational students – Limited English language skills will not be a barrier to admittance to and participation in Vocational Educational Programs.”

**Student Honesty Policy Reference Statement:** “Students are expected to exercise academic honesty and integrity. Violations such as cheating and plagiarism will result in disciplinary action which may include recommendation for dismissal.”

**Course Contents**

|  |  |
| --- | --- |
| **Week** | **Topic** |
| Week 01 | Output, Comments, Input, and Variables |
| Week 02 | Selection (if / elif / else) |
| Week 03 | Practice with Selection |
| Week 04 | Repetition (for, while) |
| Week 05 | Lists, Strings, and Random Numbers |
| Week 06 | Functions |
| Week 07 | Recursion, Sorting, CodingBat |
| Week 08 | Midterm Review |
| Week 09 | Midterm |
| Spr Break | Spring Break |
| Week 10 | Files |
| Week 11 | Data Mine and Import Your Own Files |
| Week 12 | Classes and OOP |
| Week 13 | Extra Credit (GUI) and Final Exam |
| Week 14 | Extra Credit (Hashing, Dictionaries) |
| Week 15 | Extra Credit (Regex, Localization) |
| Week 16 | Catch Up, Turn in Final Exam |