

Credit Hours: 3

Contact Hours: This is a 3-credit course, offered in accelerated format. This means that 16 weeks of material is covered in 8 weeks. The exact number of hours per week that you can expect to spend on each course will vary based upon the weekly coursework, as well as your study style and preferences. You should plan to spend 14-20 hours per week in each course reading material, interacting on the discussion boards, writing papers, completing projects, and doing research.

Faculty Information: Faculty contact information and office hours can be found on the faculty profile page.

COURSE DESCRIPTION AND OUTCOMES

Course Description:

This course is a comprehensive introduction to computer algorithms focusing on the analysis of various algorithms for overall efficiency. Students will develop an understanding of the use of algorithms in studying the inter-relation of the functional components, characteristics and performance of computer systems.

Additionally, students will develop an understanding of the importance of ways to solve a business problem using algorithm design and development.

Course Overview:

Welcome to the world of algorithms! This course will teach you the basic concepts for the design and analysis of algorithms, which you will use throughout your study of computer science, and throughout your career. These concepts are most effectively used through practice, and that is the approach we will take in this course. Topics we will cover include sorting, search trees, and heaps.

Course Learning Outcomes:

1. Design algorithms for solving a specific problem.
2. Implement data structures for solving a specific problem.
3. Identify data structures that can be used for solving a specific problem.
4. Discuss the factors that influence the efficiency of an algorithm.
5. Implement an algorithm that can be used to solve a specific business problem.

PARTICIPATION & ATTENDANCE

Prompt and consistent attendance in your online courses is essential for your success at CSU-Global Campus. Failure to verify your attendance within the first 7 days of this course may result in your withdrawal. If for some reason you would like to drop a course, please contact your advisor.

Online classes have deadlines, assignments, and participation requirements just like on-campus classes. Budget your time carefully and keep an open line of communication with your instructor. If you are having technical problems, problems with your assignments, or other problems that are impeding your progress, let your instructor know as soon as possible.

COURSE MATERIALS

Required:

Bouras, A. & Ainarozidou, L. (2015). *Python and algorithmic thinking for the complete beginner. Learn to think like a programmer.* Amazon Digital Services LLC. ISBN: 978-1514802168

Erwig, M. (2017). *Once upon an algorithm: how stories explain computing.* The MIT Press. ISBN: 978-0262036634

NOTE: All non-textbook required readings and materials necessary to complete assignments, discussions, and/or supplemental or required exercises are provided within the course itself. Please read through each course module carefully.

COURSE SCHEDULE

Due Dates

The Academic Week at CSU-Global begins on Monday and ends the following Sunday.

- **Discussion Boards:** The original post must be completed by Thursday at 11:59 p.m. MT and peer responses posted by Sunday at 11:59 p.m. MT. Late posts may not be awarded points.
- **Opening Exercises:** Take the Opening Exercise before reading each week's content to see which areas you will need to focus on. You may take these exercises as many times as you need. The Opening Exercises will not affect your final grade.
- **Mastery Exercises:** Students may access and retake Mastery Exercises through the last day of class until they achieve the scores they desire.
- **Critical Thinking:** Assignments are due Sunday at 11:59 p.m. MT.

WEEKLY READING AND ASSIGNMENT DETAILS

Module 1

Readings

- Chapters 1, 2, & 3 in *Python and Algorithmic Thinking for the Complete Beginner*
- Part 1 – Hansel and Gretel in *Once Upon an Algorithm*

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (50 points)

Getting Started with Python

Install Eclipse and PyDev to ensure that you have an IDE. Take screenshots of the IDE installed and ready to work. Once this is completed, create a simple program. Using draw.io, create a flowchart for your program and write pseudo-code to this. This program should accept an input, perform a mathematical manipulation to it with a defined constant variable, and then output the result. Screenshot the code and the result for completion of this assignment.

It is strongly encouraged that you submit all assignments to the TurnItIn Originality Check prior to submitting it to your instructor for grading. If you are unsure how to submit an assignment to the Originality Check tool, review the TurnItIn Originality Check – Student Guide for step-by-step instructions.

Portfolio Project Reminder (0 points)

Be sure to review the Portfolio Project Description and Portfolio Project Rubric in the Module 8 folder for details.

Module 2

Readings

- Chapters 4, 5, & 6 in *Python and Algorithmic Thinking for the Complete Beginner*
- Part 1 – Sherlock Holmes in *Once Upon an Algorithm*.

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (60 points)

Making Tea

Open up to page 51 in the *Python and Algorithmic Thinking for the Complete Beginner* text. Look at the algorithm for making tea. Discuss yesterday in terms of an algorithm. It can be the same step by step method outlined here. Try to be as detailed as possible. Are there any repetitive actions there? After you have completed this, try to create a pseudo-code scenario for your day. Think about how this might come across in code.

Write a small python segment that asks a question, reads in a response, and outputs some data that quantifies that response. Submit both screenshots of the successful code block execution and your code in a Word document for your instructor.

Module 3

Readings

- Chapters 7, 8, & 9 in *Python and Algorithmic Thinking for the Complete Beginner*
- Part 1 – Indiana Jones Part 5 only in *Once Upon an Algorithm*
- Edraw. (n.d) Explain algorithm and flowchart with examples. Retrieved from <https://www.edrawsoft.com/explain-algorithm-flowchart.php>
- Dry run testing and trace tables. (2015, February 15). Retrieved from <http://www.101computing.net/trace-table/>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (60 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Directions to the Mall

A stranger stops you on the street and asks for directions to a shop in the downtown area. This person would like both vehicle directions from the mall parking structure and walking instructions from the current location. Assume $\frac{1}{4}$ of a mile distance between the current location and the parking area. Think about providing multiple routes and times. Provide a flowchart and a trace table for directions with replacements for traffic, one-way streets, crosswalks, etc., that would need to be traversed. Your goal state is to showcase the fastest route method for the individual given several options. Create a UML diagram in and pseudo-code for your directions. Use Draw.IO to assist you with a flowchart and diagram.

Option #2: Family Secret Recipe

A family member has come to you and asked for the secret recipe to the family Chocolate Chip Cookies. There are three different recipes that your aunt, mother, and grandmother used. Create a trace table and flowchart for making the cookies. Assume all recipes preheat the oven to 375 degrees and bake for 9-12 minutes. Allow two minutes for cool down. Your goal is to showcase the differences in trace tables versus flowchart for creation of the three cookie types. Discuss why you think one cookie would be superior based upon the available information as well as the overlaps for the algorithm of creation of the cookies. Create a UML diagram in and pseudo-code for the winner. Use Draw.IO to assist you with a flowchart and diagram.

Mother

- $\frac{3}{4}$ cup granulated sugar
- $\frac{3}{4}$ cup packed brown sugar
- 1 cup butter or margarine, softened
- 1 teaspoon vanilla
- 1 egg
- 2 $\frac{1}{4}$ cups Gold Medal™ all-purpose flour
- 1 teaspoon baking soda
- $\frac{1}{2}$ teaspoon salt
- 1 cup coarsely chopped nuts
- 1 package (12 ounces) semisweet chocolate chips (2 cups)

Grandmother

- 2 $\frac{1}{4}$ cups all-purpose flour
- 1 teaspoon baking soda
- 1 teaspoon salt
- 1 cup (2 sticks) butter, softened
- $\frac{3}{4}$ cup granulated sugar
- $\frac{3}{4}$ cup packed brown sugar
- 1 teaspoon vanilla extract

2 large eggs
2 cups (12-oz. pkg.) NESTLÉ® TOLL HOUSE® Semi-Sweet Chocolate Morsels
1 cup chopped nuts

Aunt

1/2 cup (113 g) butter
½ cup (99 g) granulated sugar
¼ cup (54 g) brown sugar packed
2 teaspoons vanilla extract
1 large egg
1 ¾ cups (210 g) all-purpose flour*
½ teaspoon baking soda
½ teaspoon kosher salt
1 cup (170 g) semisweet chocolate chips

Module 4

Readings

- Chapters 15, 16, 17, & 18 in *Python and Algorithmic Thinking for the Complete Beginner*
- Part 2 – Groundhog day *Once Upon an Algorithm*

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (65 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Broken Down Car

You are out on your own and you have come into the sad issue of the old car (20+ years old, 150k miles) your parents gave you having a blown transmission unexpectedly. (It really wasn't. You just put it off because it's not cheap.) You now have the dubious choice to purchase a new\used car **OR** to repair the car your parents gave you free and clear. Assume you have a part-time job making \$10 an hour at 30 hours a week and a 15% tax rate. Your total living expenses are \$700 a month. The repairs for a transmission at \$2,500 dollars, which you can finance at 17% interest. A new vehicle would be \$20,000 at 10% interest, and a used vehicle would be \$9,000 at 12% interest with unknown driver history. Create a flowchart, trace-table, and UML for these options and then discuss which you would choose and why. Discuss affordability, reliability, and return on investment.

Complete the python exercise in the *Python and Algorithmic Thinking for the Complete Beginner* text. text, page 310 number 14.

Option #2: Date Night Dilemma

You have recently asked someone on a date. As you are trying to be impressive, you are looking to fund the entire experience yourself. You have \$150 to spend. There are several available options. The latest

large screen film presentation of the hot blockbuster superhero movie is playing at 7 PM. This would be \$65 dollars. The local DaVinci ristorante is serving the town's best food. That would be \$110 dollars with tip. There is a Shakespeare in the park presentation of *Much Ado About Nothing* for \$45 dollars from 4 PM to 6 PM. Star Light Lanes has announced a happy hour Cosmic Bowl-a-Rama for \$30 dollars going from 9-Midnight. There is a Disco Skate festival at the Madhouse Skate Rink from 7-10 PM. Fancy-dogs will get you a custom gourmet hot-dog and tornado fries for \$15 a person. Ten Guys does burgers of varying sizes with fries and a drink for about \$12 dollars a person. Fukiyama Sushi does an all you can eat sushi buffet for \$40 dollars a person. There is a moonlight boat trip starting at 10 PM for \$35 dollars a person. Given these options, create a flowchart, trace table and UML for your best date. Discuss what you would choose based upon affordability, reliability, and what your goals were for the date.

Complete the python exercise in the *Python and Algorithmic Thinking for the Complete Beginner* text, page 310 number 13.

Module 5

Readings

- Chapters 23, 24 & 25 Python in *Algorithmic Thinking for the Complete Beginner*
- Part 2 – Groundhog day in *Once Upon an Algorithm*

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (65 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Pumping Gas

A gas station attendant in the 1950's performs a series of actions for every customer that comes in. The list is as follows.

1. greet customer
2. ask for required type of gas and amount
3. ask whether customer needs other services
4. ask for required amount of money
5. give money to cashier
6. wait for change and receipt
7. give change and receipt to customer
8. say thank you and goodbye.

He does not follow these actions when he is not on shift or when there is no customer to serve. There should also be instructions for when to stop performing an action. Think about how this might be created in a computer program. Create a trace table, flowchart, and UML diagram for the above scenario.

Write a program in Python which uses a while loop to sum the squares of integers (starting from 1) until the total exceeds 200. Print the final total and the last number to be squared and added.

Option #2: Beach Vacation

Today is the day to go to the beach. You want to get from home to Mission Viejo in San Diego. You could, as an example, not a rule:

- drive for exactly 15 minutes. After 15 minutes, stop the car and get out.
- drive for exactly 8km. After 8km, stop the car and get out.
- drive as long as you are not in Mission Viejo. When you arrive, stop the car and get out.

These could allow you to arrive in Mission Viejo. However, none of them could get you to Mission Viejo. You could encounter traffic, heavy or light. You could fall short of your destination or overshoot it, depending upon your starting distance. You may hit a detour or roadblock. Include instructions for checks of 'Am I at the beach?' in your looping statement and provide enough instruction in the statement to get you to Mission Viejo.

Write a program which sums the integers from 1 to 10 using a for loop (and prints the total at the end).

Module 6

Readings

- Chapters 38 & 39 in *Python and Algorithmic Thinking for the Complete Beginner*
- Part 2 – Back to the Future in *Once Upon an Algorithm*
- Awasthi, A. (2018, March 27). Thinking recursively in python. Retrieved from <https://realpython.com/python-thinking-recursively/>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (70 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Palindrome

Recursion is a powerful tool. In python, write a recursive function that computes whether or not an array, string or integer, is a palindrome. Remember that a palindrome is identical forwards or backwards.

Examples:

Tacocat

12345654321

Take in information from a user and store it into an array. There is no size limitation to the palindrome. Ensure that you input checks for the palindrome and output information to the user if the array stored is not a palindrome.

Option #2: Minimum Element in an Array

Recursion is a powerful tool. In python, write a recursive function that compute the minimum element in an array. Rather than get a series of elements from the user, build a random number generator and have the user designate the number of elements to be put into the array. Once that is selected, use a for loop to generate the number of elements to be stored into the array and use a recursive function to go through the array and locate the smallest element. Ensure you put forth error messages for any input that is not an integer.

Module 7

Readings

- Part 2 – Harry Potter in *Once Upon an Algorithm*
- Foxwell, M. (2012, April 26) Coding efficiency. Retrieved from https://www.phusewiki.org/wiki/index.php?title=Coding_Efficiency

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Module 8

Readings

- Chapters 34 & 35 in *Python and Algorithmic Thinking for the Complete Beginner*
- Part 1 – Indiana Jones Parts 6 & 7 only in *Once Upon an Algorithm*
- Snyder, W. (2017, Fall). Growth rate of functions and the complexity of algorithms. *Boston University*. Retrieved from <http://www.cs.bu.edu/~snyder/cs109/CS109.MathLect04.ComplexityAlgorithms.pdf>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Portfolio Project (350 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Reflection and Revision

Submit your flowcharts, tables, and code from Modules 2-6 with corrections. Be sure to highlight your changes and why you made these changes to your work.

Instructions:

- Your paper should be three to four pages long, or approximately 300 words (excluding cover page and references).
- Your responses should be well supported.
- Your paper must include a minimum of two credible, non-text references and be formatted according to *CSU Global Guide to Writing & APA*.
- Use the CSU-Global library to find references.
- Be sure to discuss and reference concepts taken from the assigned textbook reading and relevant research.

Option #2: Reflection and Redesign

Select two of your programs from the previous mods. Using the techniques you have learned up to now, redesign them and make them more complex. Be sure to create a new flowchart and table. In an essay, discuss the changes that you made to the programs and why you made these changes.

Instructions:

- Your paper should be three to four pages long, or approximately 300 words (excluding cover page and references).
- Your responses should be well supported.
- Your paper must include a minimum of two credible, non-text references and be formatted according to *CSU Global Guide to Writing & APA*.
- Use the CSU-Global library to find references.
Be sure to discuss and reference concepts taken from the assigned textbook reading and relevant research.

COURSE POLICIES

Grading Scale	
A	95.0 – 100
A-	90.0 – 94.9
B+	86.7 – 89.9
B	83.3 – 86.6
B-	80.0 – 83.2
C+	75.0 – 79.9
C	70.0 – 74.9
D	60.0 – 69.9
F	59.9 or below

Course Grading

20% Discussion Participation
0% Opening Exercises
8% Mastery Exercises
37% Critical Thinking Assignments
35% Final Portfolio Project

IN-CLASSROOM POLICIES

For information on late work and incomplete grade policies, please refer to our [In-Classroom Student Policies and Guidelines](#) or the Academic Catalog for comprehensive documentation of CSU-Global institutional policies.

Academic Integrity

Students must assume responsibility for maintaining honesty in all work submitted for credit and in any other work designated by the instructor of the course. Academic dishonesty includes cheating, fabrication, facilitating academic dishonesty, plagiarism, reusing /repurposing your own work (see CSU-Global Guide to Writing & APA for percentage of repurposed work that can be used in an assignment), unauthorized possession of academic materials, and unauthorized collaboration. The CSU-Global Library provides information on how students can avoid plagiarism by understanding what it is and how to use the Library and internet resources.

Citing Sources with APA Style

All students are expected to follow the CSU-Global Guide to Writing & APA when citing in APA (based on the most recent APA style manual) for all assignments. A link to this guide should also be provided within most assignment descriptions in your course.

Disability Services Statement

CSU-Global is committed to providing reasonable accommodations for all persons with disabilities. Any student with a documented disability requesting academic accommodations should contact the Disability Resource Coordinator at 720-279-0650 and/or email ada@CSUGlobal.edu for additional information to coordinate reasonable accommodations for students with documented disabilities.

Netiquette

Respect the diversity of opinions among the instructor and classmates and engage with them in a courteous, respectful, and professional manner. All posts and classroom communication must be conducted in accordance with the student code of conduct. Think before you push the Send button. Did you say just what you meant? How will the person on the other end read the words?

Maintain an environment free of harassment, stalking, threats, abuse, insults, or humiliation toward the instructor and classmates. This includes, but is not limited to, demeaning written or oral comments of an ethnic, religious, age, disability, sexist (or sexual orientation), or racist nature; and the unwanted sexual advances or intimidations by email, or on discussion boards and other postings within or connected to the online classroom. If you have concerns about something that has been said, please let your instructor know.