Advanced Placement Computer Science Principles

2022-2023

Instructor: *Joshua Bas*
email: joshua.bas@southlakechristian.org

Mission Statement: SouthLake Christian Academy exists for the sake of God's glory and mission; proclaiming the gospel and discipling the whole person in all aspects of God's reality.

Course Description: The AP Computer Science Principles (CSP) curriculum is a full year, rigorous, first year college entry-level course that introduces high school students to the foundations of modern computing. The course covers a broad range of foundational topics such as digital information, programming, algorithms, the Internet, big data, digital privacy and security, and the societal impacts of computing. The course is centered around five "Big Ideas". Further, six "Computational Thinking Practices" spiral throughout the "Big Ideas".

Big Ideas	Computational Thinking Practices
• Creative Development `[CRD]`	• Computational Solution Design `[CTP1]`
• Data `[DAT]`	• Algorithms and Program Development `[CTP2]`
• Algorithms and Programming `[AAP]`	• Abstraction in Program Development `[CTP3]`
• Computing Systems and Networks `[CSN]`	• Code Analysis `[CTP4]`
• Impact of Computing `[IOC]`	• Computing Innovations `[CTP5]`
	• Responsible Computing `[CTP6]`

Technology: SouthLake will issue a Macbook device to each student, which we will use daily. Much of the work we do will be in a web environment; it is recommended that students have a current internet browser loaded on their devices (recommended: Mozilla Firefox and/or Google Chrome – both are available for download free of charge.)

AP Course Assessments: AP courses are *college-level* courses. Therefore, in an effort to best prepare our students and to reflect the college credit they can potentially earn, policies for these courses will be more in line with collegiate policies and in some instances may differ from honors and college-prep classes at SLCA. Students and parents should carefully review the following policies to ensure a full understanding of the expectations of AP teachers.

Tests may include, but are not limited to, in-class tests, take home tests, major papers or projects, etc.

Quizzes may include, but are not limited to, in-class quizzes, guided reading assignments, vocabulary, etc.

Project Work: We will be using various online resources as well as "unplugged" activities. There will be

many online projects assigned that will involve data analysis, manipulating digital information and computer programming projects. These will be assessed as either test or quiz grades depending on the level of effort required.

During the Spring semester, **students are required to take the AP exam** for each AP class in which they are enrolled. This exam will be administered at SLCA on TBD. Students are registered for the exams by SLCA, but are required to pay for each exam taken. The cost is $100 per exam. Students for whom this presents a financial hardship should contact Ms. Julie Swanson. Students who do not take AP exams will have their AP course reclassified as an honors course on their transcript. AP exams are not factored into the second semester grade because The College Board grades them and scores are not available until July.

Additionally, **AP Computer Science Principles requires the submission of a Create Performance Task (Create PT) along with responses to prompts regarding the task.** The Create PT is due on TBD. Students will be provided with detailed requirements and instructions for creating and submitting the performance task and related free response questions. The Create PT accounts for 40% of the final AP score. The College Board requires that students be given 12 hours of in-class time to complete the task. Students may work on the tasks at home in addition to the time allotted in class. The 2 hour AP exam on TBD will account for 60% of the students' final AP grade.

Numerical Scoring Guide - Students will accrue points as follows:

- Independent Work assignments will be worth 1-10 points each.
- Quiz Grades will be worth 11-25 points each.
- Test Grades will be worth 26-100 points each.

Final Grade

- 80% Semester Grade
- 20% Semester Exam

Attendance/Absence Policy:

- Attendance in class is absolutely essential to the successful completion of the course and to the attainment of a qualified grade on the nation-wide Advance Placement examination held in May. The rigors of this class demand you miss a minimal number of classes.
- If a student is absent on the day a test is administered in class, s/he will have two school days after returning to make up the test. It is the student's responsibility to make testing arrangements within that timeframe. No test can be made up after the two school days have lapsed.
- If a student is absent on the day an in-class quiz is administered, they will receive a "0" for that quiz. At the end of the semester, the instructor will drop the two lowest in-class quiz grades for each student, meaning a student may miss up to two quizzes in a semester without being penalized for their absence. *NOTE:* This policy applies only to in-class quizzed and does not apply to assignments that receive credit as a quiz grade. Assignments that were not turned in on time and received a "0" are not available to drop. Assignments missed due to SLCA business may be made up either prior to or within two school days.
- Any assignment (other than an in-class test or quiz) regardless of whether it is counted as a quiz or test grade, must be turned in on the assigned due even if the student is not in class. It is the student's responsibility to either turn the work in early or see that it is submitted to the teacher (not the front

office) on the due date. Assignments not turned in on the assigned due date receive no credit, and there is no opportunity to make up these grades.

Classroom Expectations:

- Assignments are to be completed in a timely manner and in accordance with instructions.
- Participation in class is required.
- The student is expected to take full responsibility for communicating with the teacher concerning absences and excuses and completing make-up work according to SouthLake Christian policy.
- The attitudes of students are to reflect Christ in honesty, humility, and honor.

Honor Code: Students will abide by the honor code statement ***"I shall neither give nor receive help"*** for all assignments unless specifically exempted by the instructor. Violations of the honor code pledge will result in a zero for the assignment; actions will follow the school's policy on cheating. The following table is an **inexhaustive** list of reasonable and non-reasonable behaviors:

Reasonable Behavior	Non-reasonable Behavior
• Discussing course materials and concepts with the teacher to gain deeper understanding	• Searching for solutions to a problem before the submission date
• Discussing course materials and concepts with other students to gain deeper understanding	• Asking for a classmate's solution to a problem before the submission date
• Discussing at a high-level solutions to problems using diagrams or pseudocode, but not actual code	• Giving a classmate your solution to a problem before the submission date
• Discussing at a low-level solutions to problems using actual code **provided that the problem is part of group work and you are discussing with your partner(s)**	• Using others' code segments in more than 10% of your work, regardless if you cited properly
• Using language references (online or otherwise) or searching for solutions to *technical difficulties* (e.g. your Python path has been corrupted)	• Looking at others' work during a quiz or test
• Working with (and paying) a tutor to help you with concepts, **provided that the tutor does not provide the solutions to problems**	• Attempt to access problem sets or problem solutions before those problems are published
• Copying program segments with proper citation	• Paying or offering to pay someone to do your work for you
	• Splitting the workload and combining solutions (i.e. you must attempt each part of a problem, regardless if working with a partner)
	• Attempting to exploit software bugs in any of the resources or platforms used

If there is any question that your behavior may be non-reasonable, please come to me for guidance first!

Required Supplies:

- School-issued Macbooks, **charged**
- Pencils
- Notebook / 3-ring binder

Resources:

- replit.com
- Dale, Nell, and John Lewis. *Computer Science Illuminated*. Jones & Bartlett Learning, 2016.
- Matthes, Eric. *Python Crash Course – A Hands-on, Project-Based Introduction to Programming*. No Starch Press, 2016.
- Shiffman, Daniel. *The Nature of Code: Simulating Natural Systems with Processing*, D. Shiffman, 2012. *The Nature of Code*, natureofcode.com/book/introduction/.

Curriculum Layout [*subject to change*]:

Week	Unit	Content	Big Idea
3	Unit 0: Representing Information	• abstraction • binary conversion • 2's complement vs unsigned • binary arithmetic • images • analog vs digital • sampling • lossless vs lossy compression	`CRD, DAT`

- **Unit 0 Activity for *Black & White and Color Images***
  - Students will work in pairs to create a bitmap that corresponds to a black and white image in a series of images.
  - Students will work in pairs to create a bitmap that corresponds to a 24-bit color image in a series of images.
- **Unit 0 Activity for *Text Compression***
  - Students are given a paragraph and work in pairs to analyze the text
  - Students work in pairs to create a lossless compression protocol using a dictionary
  - Students calculate the compression rate
  - Students repeat the activity, instead creating a lossy compression protocol
  - Students compare and contrast their two protocols
- **Unit 0 Activity for *Extension: Image Compression***
  - Students will discuss in large groups how they might compress images
  - Students will discuss in large groups how image compression might be similar/different than text compression
  - Using the discussion results, students will compress a simple black and white image via a handout

---

2	Unit 1: Communicating Information	• the internet • routing • packets • protocol • IP, TCP, UDP, HTTP, DNS • fault tolerance • sequential vs parallel computing	DAT, CSN, IOC

- **Unit 1 Activity for *Communicating Information***
  - Students will pick a real-world task (e.g. brush teeth, walk to the door, etc.) and create a flowchart to diagram the steps to complete such a task. This activity is a soft-introduction to algorithms and a hard-introduction to the specificity computers need in order to communicate information.
- **Unit 1 Activity for *Routing***
  - Orange Activity from CS Unplugged
- **Unit 1 Activity for *Parallel Computing*** `[CTP1, CTP2]`
  - Students work in pairs to diagram a real-world task whose steps can be done sequentially and whose steps can be done in parallel. Students will then calculate how long each variation of the task takes.
  - Students discuss in large groups the pros and cons of sequential vs parallel computing.

2	Unit 2: Programming – Variables, Conditionals, and Functions		`CRD, AAP`

- **Unit 2 Activity for *Variables, `input()`, & `print()`*** `[CTP1, CTP2]`
  - Student will work in pairs to program a simple cash register, where the program must calculate the total purchase cost of 3 items with a 7.5% sales tax.

3	Unit 3: Programming – Graphics	• using the `turtle` module • using the `apcsp` module	`CRD, AAP`

•

4	Unit 4: Programming – Loops and Lists		`CRD, AAP`

- **Unit 4 Activity for *Lists and Loops*** `[CTP1, CTP2]`
  - Building off their previous work, student will work in pairs to program a simple cash register, where the program must calculate the total purchase cost of an arbitrary number of items with a variable sales tax using lists to store item names and prices.
  - The base program features the procedural abstractions for the following functions, which must be implemented: `addItem`, `removeItem`, `computeSubtotal`, and `computeTotal`.
- **Unit 4 Activity for *Practice Create Performance Task*** `[CTP1, CTP2, CTP3, CTP4, CTP6]`
  - Students will work in pairs to design, diagram, implement, and test a complex program that has purpose in their personal lives. This program must use lists and loops.
  - Students will cite any external code segments or APIs used.
  - Students will work in pairs to answer prompts about their program.

3	Unit 5: Algorithms and Simulations	• sorting algorithms • search algorithms	`AAP, IOC`

- **Unit 4 Activity for *Genetic Algorithm***
  - Students will implement the `fitness()` and `mutate()` functions of a larger genetic algorithm to simulate evolutionary behavior, following the examples in Daniel Shiffman's *Nature of Code*.
  - Students will analyze, comment, and compare various code segments throughout the program.
  - Students will discuss whether genetic algorithms are heuristics or optimal solutions.
- **Unit 4 Activity for *Extension: Implementing Binary Search*** `[CTP4]`
  - Students will compare and contrast linear search to binary search using the `time` module.
  - Students will work in pairs (driver and navigator) to implement binary search.
  - Students will create unit tests of their binary search implementation.

4	Unit 6: Computing in the Real World		`CRD, IOC`

•
