SECONDARY COURSE DESCRIPTION

SECTION A: COURSE CLASSIFICATION

ALAMEDA UNIFIED SCHOOL DISTRICT
Excellence & Equity For All Students

1. Course Title: Computer Science Discoveries CTE 1	6. Prerequisite(s): None
2. Action: _X_ New Course _ ___ Course Revision ___ Title Change Only	7. Grade Level: 9 - 11
3. Transcript Title/Abbreviation: CS Discovery 1 (For Educational Services)	8. Elective/Required: CTE Pathway Elective (1 year course)
4. Transcript Course Code/Course Number: VQJI (For Educational Services)	9. Subject Area: CTE
5. CBEDS Code: (For Educational Services)	10. Department: CTE

11. Length /Credits:
___ 0.5 (half year or semester equivalent) _x__ 1.0 (one year equivalent)
___ 2.0 (two year equivalent)

12. Was this course previously approved by UC? ___ Yes ___x No
If so, year removed from list: _____

13. Meets the "__D___" requirements in the a-g university/college entrance requirement. Approval date:
5/9/19

14. School Contact Information
Name: _____Nancy Read_____ Title/Position: __ICT CTE Teacher_____
Phone: _____ Fax: _____
E-Mail: __nread@alameda.k12.ca.us_____

16. Signatures:
Department Chair: Nancy Read
Principal: Robert D.
Acknowledged by Other Principals:
Educational Services:

16. BOE Approval
Signature of Superintendent:_____ Date of Approval_____

17. Course Description: Computer Science Discoveries (CS Discoveries) is an introductory computer science course that empowers students to create authentic artifacts and engage with computer science as a medium for creativity, communication, problem solving, and fun. CS Discoveries is designed from the ground up to be an accessible and engaging course for all students, regardless of background or prior experience. It provides students opportunities to engage with culturally and personally relevant topics in a wide variety of contexts and aims to show all students that CS is for them. This course covers the following concepts:
    - Problem solving
    - Web development
    - Animation and games
    - The design process
    - Data and society
    - Physical computing

18. Course Goals and/or Major Student Outcomes:
    Computer Science encompasses far more than just coding, and CS Discoveries will provide students with opportunities to explore the many facets of CS, both in terms of how they are personally relevant as well as how they impact society. This course introduces students to computer science as a vehicle for problem solving, communication, and personal expression. It focuses on the visible aspects of computing and computer science, and encourages students to see where computer science exists around them and how they can engage with it as a tool for exploration and expression. Additionally, this course asks students to look outward and explore the impact of computer science on society. Students will see how a thorough user-centered design process produces a better application, how data is used to address problems that affect large numbers of people, and how physical computing with bare circuit boards allows computers to collect input and return output in a variety of ways. Finally, as a CTE course, students explore career options involving technology and tech companies.

19. Course Objectives (standards): Students will be able to:
    - Explain how computers input, output, store, and process information to help humans solve problems
    - Design an application that helps solve a problem
    - Design and build a website using HTML, java script, and CSS
    - Express valuable programming skills such as debugging, commenting, and structure of language
    - Express the impact of sharing information online and how to be more critical content consumers
    - Identify existing and potential cybersecurity concerns and options to address them
    - Demonstrate programming concepts and the design process computer scientists use daily by creating computer games
    - Collaborate when processing information to gain insight and knowledge
    - Express an algorithm in a programming language
    - Explain how people participate in a problem-solving process that scales
    - Explain how computing has impacted innovations in other fields
    - Engage with a team to identify and solve problems

20. Course Outline:

Semester 1 - Exploration and Expression

The first semester of CS Discoveries introduces students to computer science as a vehicle for problem solving, communication, and personal expression. As a whole, this semester focuses on the visible aspects of computing and computer science, and encourages students to see where computer science exists around them and how they can engage with it as a tool for exploration and expression.

Unit 1 - Problem Solving

Unit 1 is a highly interactive and collaborative introduction to the field of computer science, as framed within the broader pursuit of solving problems. Through a series of puzzles, challenges, and real world scenarios, students are introduced to a problem solving process that they will return to repeatedly throughout the course. Students then learn how computers input, output, store, and process information to help humans solve problems. The unit concludes with students designing an application that helps solve a problem of their choosing.

CTE Standards:
Communication: 2.5; Technology: 4.3; Problem Solving: 5.1-2, 5.4-7, 5.9; Leadership & Teamwork: 9.2-2, 9.6; Technical Knowledge & Skill: 10.5-6, 10.10; Develop Web and online projects C7.1, C7.5; Create effective interfaces between humans and technology: C3.1-3

Unit 2 - Web Development

In Unit 2, students are empowered to create and share the content on their own web pages. They begin by thinking about the role of the web, and how it can be used as a medium for creative expression. As students develop their pages and begin to see themselves as programmers, they are encouraged think critically about the impact of sharing information online and how to be more critical content consumers. They are also introduced to problem solving as it relates to programming, as they learn valuable skills such as debugging, commenting, and structure of language. At the conclusion of the unit, students compile their work to create a personal website they can publish and share.

CTE Standards:
Communication: 2.3; Technology: 4.3-4; Problem Solving & Critical Thinking: 5.1, 5.3-5, 5.8; Responsibility: 7.-5; Ethics & Responsibility: 8.3, 8.6-7; Leadership & Teamwork: 9.7; Develop software using programming languages: C4.3, C4.5, C4.6; Integrate a variety of media into development projects: C6.1, C6.6, C6.7; Develop software using programming languages C4.3, C4.11; Test, debug, and improve software development work: C5.3-6;

Unit 3 - Animations and Games

In Unit 3, students build on their coding experience as they create programmatic images, animations, interactive art, and games. Starting off with simple, primitive shapes and building up to more sophisticated sprite-based games, students become familiar with the programming concepts and the design process computer scientists use daily. They then learn how these simpler constructs can be combined to create more complex programs. In the final project, students develop a personalized, interactive program. Along the way, they practice design, testing, and iteration, as they come to see that failure and debugging are an expected and valuable part of the programming process.

CTE Standards:
Communication: 2.3; Problem Solving & Critical Thinking: 5.1, 5.3-5, 5.8; Responsibility: 7.-5; Leadership & Teamwork: 9.2-3, 9.7; Identify and apply the systems development process: C1.4; Develop software using programming languages: C4.3, C4.5, C4.6; Test, debug, and improve software development work: C5.1-5.6; Integrate a variety of media into development projects: C6.1, C6.6, C6.7;

Semester 2 - Innovation and Impact

Where the first semester centers on the immediately observable and personally applicable elements of computer science, the second semester asks students to look outward and explore the impact of computer science on society.

Students will see how a thorough user-centered design process produces a better application, how data is used to address problems that affect large numbers of people, and how physical computing with bare circuit boards allows computers to collect input and return output in a variety of ways.

Unit 4 - The Design Process

Unit 4 transitions students from thinking about computer science as a tool to solve their own problems towards considering the broader social impacts of computing. Through a series of design challenges, students are asked to consider and understand the needs of others while developing a solution to a problem. The second half of the unit consists of an iterative team project, during which students have the opportunity to identify a need that they care about, prototype solutions both on paper and in App Lab, and test their solutions with real users to get feedback and drive further iteration.

CTE Standards:
Communication: 2.3; Problem Solving & Critical Thinking: 5.1-5.12; Leadership & Teamwork: 9.2-3, 9.7; Identify and apply the systems development process: C1.4-1.5; Define and analyze systems and software requirements: Develop software using programming languages: C4.3, C4.5, C4.6; Test, debug, and improve software development work: C5.1-5.6; Integrate a variety of media into development projects: C6.1, C6.6, C6.7;

Unit 5 - Data and Society

This unit is about the importance of data in solving problems and highlights how computers can help in this process. The first chapter explores different systems used to represent information in a computer and the challenges and tradeoffs posed by using them. In the second chapter students learn how collections of data are used to solve problems, and how computers help to automate the steps of this process. The chapter concludes by considering how the data problem solving process can be applied to an area of the students' choosing.

CTE Standards:

Communication: 2.3; Develop software using programming languages: C4.1, C4.6-7
Problem Solving & Critical Thinking: 5.1, 5.3-5, 5.8; Responsibility: 7.5; Leadership & Teamwork: 9.2-3, 9.7; Identify and apply the systems development process: C1.1, C1.4-1.5; Test, debug, and improve software development work: C5.4-5.6; Integrate a variety of media into development projects: C6.1, C6.6, C6.7;

Unit 6 - Physical Computing

In Unit 6, students further develop their programming skills while exploring the role of hardware platforms in computing. Harkening back to the Input and Output elements of the Input/Storage/Processing/Output model for a computing, students look towards current and "smart" devices to understand the ways in which different sensors can provide more effective input and output than the traditional keyboard, mouse, and monitor. Using App Lab and Adafruit's Circuit Playground, students develop programs that utilize the same hardware inputs and outputs that students saw in the smart devices they explored earlier, and they get to see how a simple rough prototype can lead to a finished product. The unit concludes with a design challenge that asks students to use the Circuit Playground as the basis for an innovation of their own design.

CTE Standards:
Communication: 2.3-5; Problem Solving & Critical Thinking: 5.1-5.12; Leadership & Teamwork: 9.2-3, 9.7; Identify and apply the systems development process: C1.4; Create effective interfaces between humans and technology: C3.1-3.3; Develop software using programming languages: C4.3, C4.5, C4.6, C4.9, 4.11; Test, debug, and improve software development work: C5.1-5.6; Integrate a variety of media into development projects: C6.6; Develop software for a variety of devices, including robotics: C9.1-C9.3, C9.5

21. Instructional Materials:
    Board approved required text:

Supplementary materials:

CODE.org provides all teacher lesson plans as well as student side activities within their free software for lessons, the game lab and app lab. Please see https://code.org/educate/csd for more details on the course. The course does require a set of Circuit Playground modules, one for each pair of students. These modules are offered at discount to CS Discovery teachers. A class set costs less than $200.

22. Instructional Methods and/or Strategies

The CS Discoveries curriculum is very student focused, project based and engaging. The curriculum creates many chances for self-reflection, formative and summative assessment, and peer feedback. A wide variety of instructional methodologies and strategies are utilized to meet the needs of diverse learners and successfully support the learning goals. Some of these include:

- Team-based: all programming is done in pairs. Working in pairs allows stronger students to reinforce their knowledge by explaining their rational while less-expert students are supported not only by their teacher, but with their partner.
- Instructor uses a variety of technology tools, like videos and guided notes to support lectures and the introduction of new materials
- Regular formative and summative assessments are built into the curriculum to ensure that all students are synthesizing the material. Struggling students are given individual or small group assistance
- SPED/ELD students often thrive in a computer-based environment where they can utilize translation tools, engage in hands on and interactive exercises where they get immediate feedback for their work. Curriculum can be modified if needed to meet IEP requirements.

23. Assessment and Evaluation

The curriculum creates many chances for self-reflection, formative and summative assessment, and peer feedback with each unit. Students are required to keep journals, both on paper and digital, documenting their progress. Rubrics are given for projects. Students must work successfully in their teams and meet certain benchmarks to demonstrate that they are meeting the learning goals before progressing further in the class.

24. Grading Policy

Grades are based assignments, projects, and effort. All students keep both a digital and notebook portfolio of their assigned work. For major projects, rubrics and sample work will be provided to ensure student expectation.

SECTION C. OPTIONAL INFORMATION

25. Context for offering the course:

Computer science is quickly becoming a K-12 requirement in many districts in the Bay Area and beyond. Additionally, there are approximately 1.4 million jobs in the field of computer science and only about 400,000 people in the pipeline to fill those jobs. We really need to be preparing students for those opportunities. Finally, the cognitive, communication and collaboration skills learned in this course are beneficial for students no matter what their pathway is after high school.

The district has been experimenting with course content to develop the optimal CTE computer science pathway. We have determined that Computer Science Discoveries is the best curriculum to use to introduce computer science concepts and programming in a fund and relevant environment that is designed to reduce the stress and allow students to be more successful. Students will be well-prepared to move to the second-level course (Computer Science Principles

CTE 2 and AP Computer Science Principles CTE 2). We believe if students have access to a CS pathway prior to registering for AP Computer Science A we will see a more diverse population in that classroom.

26. History of Course Description: This course was written by CODE.org and is intended as a preparatory class for Computer Science Principles and AP Computer Science Principles.