SCHOOL of EDUCATION

INSTRUCTIONAL PLAN

Name: Gary Malkasian	**Grade: 9-12**
Date: 11/12/24	**Unit/Subject: Computer Science: Programming in C#**
Lesson: History of Computer Science	**Number if Part of a Series Lessons: 1 of 5**

Learning Target(s)

Long-term Learning Goal(s) (i.e., teacher goal for unit or learning segment):

These are the introductory lessons in writing computer programs and introduce basic computer programming terminology and concepts: data structures, variables, constants, loops, and Boolean logic. Students will learn how to create simple computer programs. This lesson on computer science history will connect their studies to the historical context.

This lesson:
Discuss how computing technologies have changed the world, prominent contributors, and the mutual impacts with society.
Evaluate ways computing has changed your life, at home, in school, with your friends, and in your environment.
Predict how future technologies could continue to change.
Follow precisely complex multistep procedures in performing programming tasks.
Determine and use correct terminology to describe programming concepts.

Common Core: Reading Standards for Literacy in Science and Technical Subjects 6–12

Craft & Structure
Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to grades 9–12 texts and topics.

OPSI Computer Science K-12 Standards

1B-IC-18 Discuss computing technologies that have changed the world, and express how those technologies influence, and are influenced by, cultural practices.
3A-IC-24 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
3B-IC-27 Predict how computational innovations that have revolutionized aspects of our culture might evolve.

Learning Target(s) for this Lesson: (in kid-friendly language): This is a lesson on the history of computing, the pioneers, and how the devices we use every day fit into that history. We will discuss how computers have evolved in society, some of the key people involved, and what the principal achievements have been.

Academic Language explicitly taught in this lesson – vocab, language function (processes such as explain, describe, analyze), syntax, discourse

Academic Language taught: algorithm, memory, storage, processing, computer science, Internet, model, processor, analog, digital, system, wearable technology, operating systems, programming, machine language, assembly language, and high level programming languages.

Academic Language Supports (charts, handouts, videos etc.): PowerPoint, Do Now handout

Personal, Cultural or Community Asset Connections

Plan (what will I do to connect the target(s) to students and how?):
The lesson connects ancient history to the modern world with lots of pictures and facts about the early pioneers.
A comprehensive look at computer science history would be a presentation about 4 times as long as this one, which would not be practical and challenging even for a college age audience, so I was selective in which people and events I would present. In particular, the accomplishments of a diverse group of pioneers is presented in the hopes that the student will see someone "that looks like me".
When selecting images, I intentionally selected images that were 1) representative of the person at the time of the achievement, and 2) were young, so the student is seeing young Ada Lovelace, young Margaret Hamilton, young Bill Gates, young Steve Jobs, etc., and think wow, these were young people doing these things.
The list of over 200 names of computer science pioneers near the end of the presentation is real and demonstrates the clear ethnic diversity of those who made these contributions.

Background Knowledge

How will I activate background knowledge?
Students will likely have heard of Bill Gates and Steve Jobs. Some may know Alan Turing or have seen the recommended films. They may also have heard of an abacus. The lesson starts with ancient, primitive tally sticks and walks through to the modern age with microprocessors, smart phones, wearable technology, and AI, making connections between history and the things they use every day.
All students will have used computers and most will have smartphones.

How will I fill prior knowledge gaps? This is introductory material. I assume no prior knowledge of computer science history or programming.

Addressing "the hard part"

What will be the "hard part" for kids about this learning target(s)? Possible misconceptions?

1. The diversity of computer science pioneers is likey to be eye-opening.
2. Students may think computer science is just for "smart people" thanks to what they learn from movies & television.
3. There is a lot of history covered in 24 PowerPoint slides – students are not expected to memorize it.
4. Some students may not want to share during discussions.
5. Some students may not be willing to make a time commitment to learning programming. I expect some will laugh or mock it.

How will I address that "hard part" (what will I do and how?)

1. PowerPoint shows many examples, and it's just the tip of the iceberg.
2. TW lead discussion on our preconceptions and encourage growth mindset
3. TW explain the presentation is an "overview", to just watch, think about, stimulate discussion, and enjoy.
4. Some of the discussion will be "popcorn style". Students can call out a word or two to give examples that haven't yet been said.
5. The commitment sheet is *for you*. I'm not collecting it. I'm asking you to bring your A game because *you respect yourself*.
   It is included in the Do Now because studies show making a commitment increases follow through.[1]
   TW ask, if you don't want to make a commitment to yourself that I'm not even going to collect, what about that frightens you?

Instructional Materials, Resources, and Technology Needed to Present this lesson

Large Screen for PowerPoint presentations and videos, Internet access, MS Teams and Canvas (or other LMS)

Instructional Plan to Deepen and Extent Student Understanding:

Teacher Tasks (steps of the lesson) and Student Learning Behavior (what you'll see)-Teacher Will or Student Will (TW or SW)- Start with an Action Word

PART I

TW greet students as they enter the classroom

SW complete "Do Now" task on their desk, answering these questions:

- What computers, or devices with computerized components, do you use?
- Describe the type of person you think of when you hear the term "computer scientist".
- Why do you want to learn computer programming?
- Fill out: I, _____ (your name), commit to spend ____ hours per week outside of class practicing code writing skills
- When during the week will you do this (days and times): _____

[1] Study showing making a written commitment to an exercise program increased follow-through from 38% to 91%: https://jamesclear.com/implementation-intentions

TW ask students about their computerized devices
SW volunteer answers "popcorn style" to see how many kinds of devices they can come up with
TW ask students about their preconceptions of computer scientists
SW will share answers – 2 to 4 students is enough

TW ask students to raise hands if they are involved in any type of sport or athletics, iterating through team sports, exercise, cheer, skiing, etc.
TW ask students to raise hands if they are involved in any type of creative arts (paint, draw, photography) or performing arts (music, theater)
TW discuss that learning to write code isn't just learning a bunch of facts, it's acquiring a skill, like sports, music, or language.
TW discuss with students:

- What does it mean to "bringing your A game" to learning to write code?
- What commitment do you make to yourself, in terms of time and effort?
- What help do you think you might need? How will you advocate for yourself?

TW announce that the students "get to decide what we'll be doing this week".

OPTION A: The teacher can give a lesson on Computer Science History today and we jump right into coding tomorrow and for the rest of the semester.

OPTION B: We can spend 2-3 days on student-led Computer Science History projects. Students will form groups, pick a subject area, and give a presentation to the class. Student presentations can be:

- PowerPoint
- Poster Board
- Short Drama (5 min)
- Show & Tell – if you have a really old computer from home
- Other artistic display (paint, sculpt, etc.)

Your presentation must illustrate the Computer Science subject area you selected.

Subject Areas:

- Early Computers: mechanical devices
- 1st Generation Digital Age: vacuum tubes, programming in machine language and assembly language
- 2nd Generation Digital Age: transistors and early programming languages
- 3rd Generation Digital Age: integrated circuits
- 4th Generation Digital Age: (now) microprocessors, personal computing, Internet

- 5th Generation Digital Age: (just starting) quantum computing, AI
- Black Computer Science Pioneers
- Female Computer Science Pioneers
- LGBTQ Computer Science Pioneers
- Other ethnic group: ask teacher

Key People and Technologies should be included in *all* presentations.

Rubric for Option B			
	Accomplished	**Emerging**	**Developing**
People	Names at least two key figures who made pivotal contributions.	Names at least one individual, but misses a central figure of greater importance.	Does not name any key individuals, or names individuals from a different category.
Technologies	For a digital generation: lists the key technologies from the era, including processor type, storage (if any), and programming language(s), and gives examples of computer systems from that generation. For a pioneer: knows in which generation the computer scientist made contributions, lists what those contributions were, and names the project or computer system impacted.	Lists key technologies, but misses central defining characteristics from that era or central contributions made by that individual.	Does not list any technologies, or misidentifies technologies from a different category.

If the students select Option B:
SW form group, select a subject area, and decide on what type of project they want to do
SW will spend the remainder of the class time researching their area and have their project as homework.

The lesson series changes if the students select Option B:

Day 1: Lesson 1: Computer Science, Part I – groups are formed and project work is begun

Day 2: Lesson 2: Introduction to Programming
Day 3: Lesson 3: Data Structures
Day 4: Lesson 1: Computer Science, Part II – group presentations, may continue for a second day if needed. At the conclusion of the presentations, TW lead class discussion as described in Part II, below.
Day 5 (or 6): Lesson 4: Operators and Statements
Day 6 (or 7): Lesson 5: Looping

If the students select Option A:
TW give lecture on the history of computer science using PowerPoint presentation and go directly to Part II. The lesson will wrap up in 1 day.

PART II

TW distribute discussion questions, listed below.
SW meet in groups to discuss the discussion questions.
While students are discussing, TW circulate to see how the groups are coming along and answer questions.
After 10-15 minutes, groups will rejoin the class. TW ask for a representative from each group, one at a time, to stand up and tell the class how their group answered one of the questions. After the representative speaks, the TW thank the student and then ask the class, "Did any group come up anything else?"

Discussion questions:

- What surprised you in the presentation(s)?
- How did the people you see differ from your original ideas about computer scientists?
  Exemplar answers:
  - There were people who looked like me
  - They looked like normal people enjoying what they do
  - Hedy Lammar OMG that's a computer scientist?
- Did anything inspire or amaze you?
  Exemplar answers:
  - Greek Antikythera: first analog computer built with gears predicted astronomical positions in 200 BC
  - The first computer programmer was a woman in 1820 (Ada Lovelace)
  - ENIAC the first digital computer in 1945 was the size of a room and took a team of programmers to manually program it by changing the electrical connections – and from the pictures, most of those programmers were women
  - A black man created the (then) fastest microchip in 1962, created multiple software companies, and used his profits to fund women- and minority-owned companies. (Frank Green)

- A woman was the director of software engineering at MIT labs in the 60s, and her team created the on-board software for NASA's Apollo project (Margaret Hamilton – no, not the one who played the Wicked Witch of the West)
- Margaret Hamilton looks like she's 12! She was the director of a NASA project? (She was 33, but yes, she was young)
- A team of black women running calculations were pivotal to the space program
- The number of devices that are computerized.
- Various diversity answers: there are prominent computer scientists that look like me, and ones that look like just about everybody I know

- What computers do you and your family use and how do you use them?

  Exemplar answers:
  - Laptop/desktops: email, homework, streaming entertainment, social media, looking up information on the Internet, games
  - Mobile phones/tablets: phone, texting, messaging, email, social media, games
  - Computerized devices: cars, appliances, home security alarm system, wearable tech
- What in your life would change if you didn't have access to any computers at all? How would you do school work, look up information, connect with friends?
- How has computing changed society?
- What do we see changing in how we compute? Do we see changes in:
  - Size of computers
  - Participation in computing – percentage of the population that has a computing device
  - Breadth of computing – types of devices that are computerized or have computerized components, types of applications
  - Increased percentage of time a person is interacting with a computer
  - Impact on society – jobs, school, social contact, culture, politics, faith traditions, hobbies & leisure
- What will the next generation of computers be like? What will we use them for? What current trends can we use to make predictions?
- What interests you in computing?

  Examplar answers:
  - Write Games
  - Web Design
  - Quantum Computing
  - Artificial Intelligence
  - Make money
  - Advocacy
  - Study computer science in college
  - (someone will say hacking) you can learn how to protect yourself from hackers, and there's "ethical hacking" which is testing a system for vulnerabilities with the permission of the owner.
- What would you like to learn in this class? What types of programs would you like to write?

Examplar answers:

- Games
- Websites
- Social media

TW note student answers and craft programming assignment options that match interests. There are limits to what can be accomplished in an introductory programming course, but simple games or scripts to run on a web page are possible.

Accommodations/Modifications Needed for Individual or Diverse Learners

- Priority seating for those requesting it.
- Lectures will be presented via MS Teams even for those in class because close captioning can be enabled in the video and it can be recorded, which can be helpful for those with language or certain learning issues.
- All PowerPoints will be available for the students.

Supports for specific special needs students for this specific target(s): PowerPoint presentation deck available to all students

This will be a presentation and discussion, with no coding. The Do Now task is the only written work, and it will not be collected. The lesson should not require as many accommodations as the more typical coding sessions.

Students with learning disabilities:
TW circulate during the group discussion time to in make sure they understood the presentation and the discussion questions, and clarify as needed. TW encourage them to participate in group discussion.

Students with ADHD:
Fidget spinners during lecture, balance balls to sit on, headphones to block sound while working are all available accommodations.

Students with Autism:
TW circulate during the group discussion time to see if they were following the presentation. TW will encourage them to participate in group discussion, but may collect answers from them directly for the discussion questions, as needed.

For ELL and all students, lessons will be recorded in MS Teams with captioning enabled. Some ELL and students with auditory processing issues may be in-class but watching the video because the captioning helps them. The recordings and PowerPoint will also be available for students to review or for those who missed class for any reason.

Assessment Strategies: Assessing the Impact of the Instruction on Student Learning	
Option A: Proposed evidence is the student responses to discussion questions. **Option B**: Proposed evidence is the student projects plus responses to discussion questions. **1B-IC-18** Discuss computing technologies that have changed the world, and express how those technologies influence, and are influenced by, cultural practices. • What do we see changing in how we compute? Do we see changes in size, participation, breadth, & impact. • What computers do you and your family use and how do you use them? • How has computing changed society? • What do we see changing in how we compute? Do we see changes in size, participation, breadth, & impact. **3A-IC-24** Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices. • What in your life would change if you didn't have access to any computers at all? How would you do school work, look up information, connect with friends? • How has computing changed society? **3B-IC-27** Predict how computational innovations that have revolutionized aspects of our culture might evolve. • What do we see changing in how we compute? Do we see changes in size, participation, breadth, & impact. • What will the next generation of computers be like? What will we use them for? What current trends can we use to make predictions?	Evaluation Criteria (What will tell me students have "met standard, etc.")? **Option B**: Conforms to Rubric above. Projects connect to subject area and have accurate information. Presentations on a computer generation identify the main features of that generation. Presentations on pioneers should include major figures and their contributions. Presentations should demonstrate the progressive improvements in computing devices, the diversity of contributions, and the impact on society. **Option A & B**: Student answers connect critical thinking skills to the presentation(s), as follows: **1B-IC-18** Discuss computing technologies that have changed the world, and express how those technologies influence, and are influenced by, cultural practices. SW recognize the tread of computers to dramatically decrease in size while increasing in computational power SW know that computers have dramatically influenced society. **3A-IC-24** Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices. SW know computer use has increased on the individual level SW know computer adoption has increased for all populations SW know application of computing has increased to touch nearly every aspect of our lives **3B-IC-27** Predict how computational innovations that have revolutionized aspects of our culture might evolve. SW predict computers will continue to shrink in size while increasing in computational power

	SW predict computerization of all human activity will continue to increase
Plan to give students feedback: TW screen projects ahead of time to correct factual errors. If there are factual errors stated during presentations, TW judge if the error is substantial enough to require a clarification to the class, and if so, do it in such a way as to not embarrass the students. SW get written feedback on projects. There are no wrong answers to the discussion questions; will give students my thoughts and ask follow-up questions, as needed. Will ask leading questions to guide discussion if no student gives an answer to meet a learning objective. For example: • If no student mentions the decrease in size, will ask "How big were the first computers? How big is your computer?" May prompt, "are you carrying a computing device in your pocket, by any chance?" • May prompt to find what groups or organizations the students are part of outside of school, and if those groups have a website, send out email, have a business computer, etc. • May ask students to imagine what they think the smallest possible computer could be.	How student will use your feedback (revise work, additional practice, etc.): • Think about how computers have changed their lives and our society • Think about the big changes in computing, and the diversity of those making important contributions • Imagine what the "next big thing" might even be • Imagine themselves making contributions in computer science