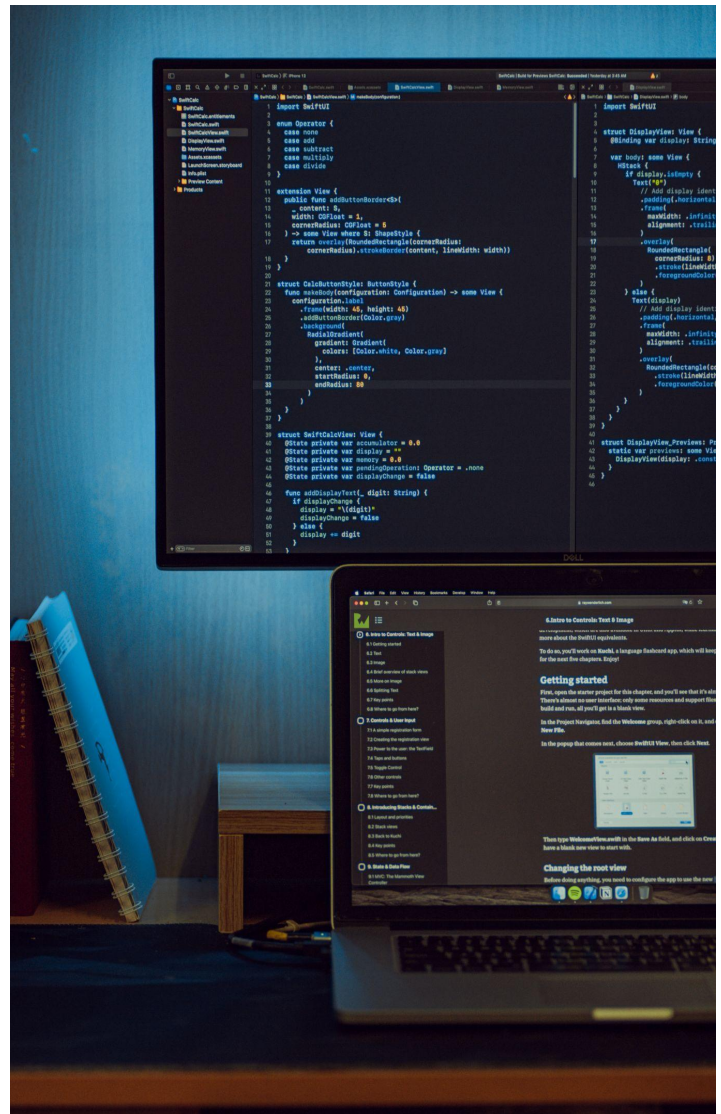


Course: Information Technology
Title: Computer Programming
Grade: 12
Topic: Arrays, Files, and Searching

Created By:
Mohammadreza
Rahimiangolkhandani
(1301459)
<https://golghandani.academichosting.ca/g12cpup>



“Be the one you love, do what you love”

Stage 1 – Desired Results

This Unit plan is designed for students in 12th grade.

Established Goal(s):

1. Students acquire the ability to categorize different programming languages based on their structure and purpose of creation
2. In each area (data, application, server), students can choose the best language for making a programming solution.
3. Students will be able to work with Array and Files and perform simple processing on them
4. Students will understand the real-world scenarios that use Files, Arrays and Search functionality
5. Students will apply their understanding to a simplified real problem

Standards:

1. National BC Standards

Students are expected to achieve the following competencies:

- Identify their perspectives and knowledge as sources of information.
- Demonstrate a sustained curiosity about the topic.
- Make observations in familiar or unfamiliar contexts.
- Make predictions about the findings of their inquiry.
- Communicate ideas, explanations, and processes in a variety of ways.

2. National Educational Technology Standards

Students are expected to be qualified to:

- Apply digital tools to gather, evaluate, and use information.
- Use critical thinking skills to plan and conduct research, manage projects, solve problems, and make decisions deliberately using appropriate digital tools and resources.
- Demonstrate creative thinking, construct knowledge, and develop innovative products and processes using technology.
- Exploit digital media and environments to communicate and work collaboratively.

Predicted Activity

- Performing visual activities by making tutorials
- Performing ELSI activities by writing articles
- Programming in their desired language
- Reviewing at least three different languages
- Reviewing at least two additional IDEs

Note: All the activities in this unit are directly related to their future career should they intend becoming a programmer.

Understanding:

1. Students would be introduced to other languages and ways of programming due to group activities. They can also decide on a future career in programming based on their own interests
2. They will learn how to write the code in a modular way
3. Documentation is important in application development because they will be asked to share their project with others and do pair programming
4. Complex tasks require different technologies and tools at different stages.

Note: there is no need to have the interface. As we know, many programs that support real-world examples are not graphical applications.

Students will know:

1. How to pick a suitable language category for each of the goals in programming,
2. The reason behind each language and the pros and cons based on their desired goal,
3. How to elicit development performance by choosing the appropriate language for the task.

Students will be able to:

1. Choose the best possible language for their next step in programming

What technologies can they use:

For article

1. MD files
2. Latex

For Video

1. iMovie for Mac
2. Media Creator Windows

For Programming

1. Visual Studio/Visual Studio Code
2. Android Studio
3. XCode
4. PyCharm

The required materials for students:

- At least a computer device (it could be a raspberry pi or a laptop/desktop computer)
- Internet Access

Requirements:

- a. Knowledge of using a computer to search for material on the internet
- b. Internet and digital literacy
- c. Previously taken Programming 11

Essential Question/Big Idea:

Importance of analyzing and applying from different sides

- a. Proper language choice (Best IDE to work with) - Critical Thinking
- b. Modular implementation - Their groupmate will give part of their whole marks

Stage 2 – Assessment Evidence

Performance Task

Goal(s):

Pick their favourite programming language based on their interests, find a real-world scenario and program a solution with their chosen language and share their path in a tutorial video or article (group activity).

Role:

Programmer and Educator (They should code the solution and make the tutorial of their implementation).

Audience:

Teacher and other classmates

(In case they would like, they can share their tutorial on social media)

Situation:

Group of 2 (Max 3) people

(Students will be asked to pair program the application) – part of their whole mark will be given by their groupmate based on how good is their cooperation as a team player.

Performance:

To begin solving real-world problems in their language of choice and later on learning how to explain their solution

Standards:

Other groups with the same chosen language should find their code modular sufficient and easy to attach a feature.

Other groups with the same chosen language should find their code clean and well documented to be easy enough to understand by reading materials.

They should present their work in a tutorial video or article (if they would like, they can share their presentation on [youtube.com](https://www.youtube.com) dev. to or medium.com).

Their documentation should contain the flow chart and code documentation

Note: 3 Rubrics for different assessments are included.

Note: Finally, they should add their projects to GitHub and present them as an open-source solution to the rest of the world.

Stage 3 – Learning Plan

W

To Understand and Apply knowledge of using Array, Files, Search in real-world applications

H

The applications that they already have in their pocket

E

Step by step, we should observe the students' performance as each step is the next foundation. Also, we can ask teammates and students with high capabilities to help others.

R

Picking a language that they are interested in. The teacher can compare different languages for the specific student and explain each one again. If the student still wants to change the language, find a group for replacement.

E

How will students self-evaluate and reflect on their learning?

Are they able to easily continue their work on the next session, or do they need to read and understand it again!

(Importance of documentation)

T

How will we tailor learning to a variety of needs, interests, and learning styles?

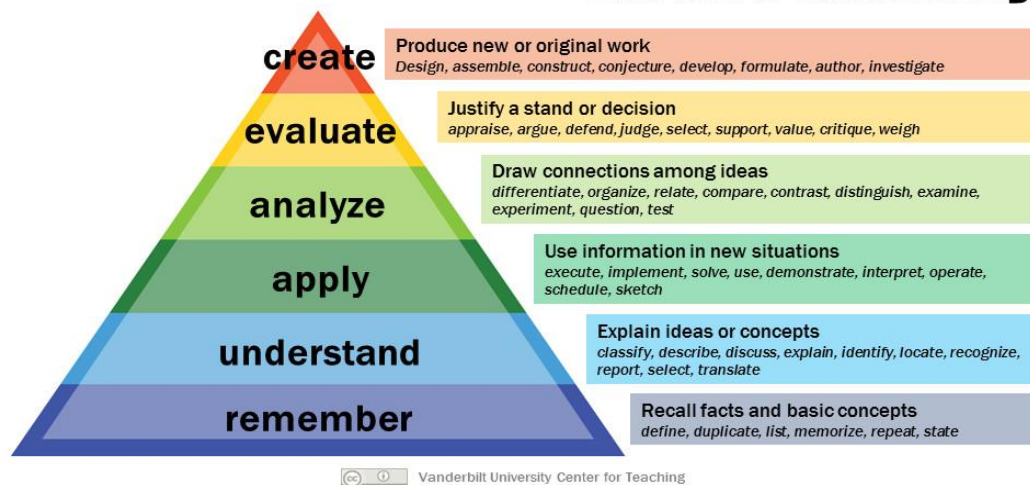
Offering different languages to work with.

There are many different ways to work on their assessments (Article, video, presentation), Peer reviewing, and peer evaluation. (Improve team-player aspects and code reading abilities)

O

The sequence of learning

Bloom's Taxonomy



Resources:

1. <https://roadmap.sh>
2. <https://www.udemy.com>
3. <https://app.pluralsight.com/paths?tab=all>
4. <https://youtube.com>
5. <https://www.khanacademy.org>
6. <https://www.freecodecamp.org>
7. <https://docs.microsoft.com/en-ca/learn/student-hub>
8. <https://github.com/ab007shetty/node-express-hbs>
9. <https://github.com/drivendata/cookiecutter-data-science>
10. <https://developer.apple.com/xcode>
11. <https://developer.android.com>
12. <https://visualstudio.microsoft.com>
13. <https://yoteachapp.com>

Learning Outcomes:

Create a real-world but simplified version of an application (No need to work with Databases) that works with Arrays, Files, and Search functionality

Note: In this Grade (12), languages should be limited to 6 languages that are trendy for each type of application

1. Java OR C# for those looking for GUI Software development on desktop (GUI should be presented)
2. Swift for those interested in iOS application development (GUI should be given)
3. Kotlin for those interested in Android application development (GUI should be provided)
4. Python for those interested in Data Analysis (Read and analyze a dataset should be provided)
5. Javascript for those interested in web development (Server/GUI should be given).

Note: The teacher should already prepare a problem for each language. For example,

1. JAVA/C# desktop application: Grocery shopping application
 - 1.1. the application should be able to get the name of an item and its price and then add it to a shopping list
 - 1.2. the application should store the data, show the whole list and calculated total price/tax, and show it to the end-user
 - 1.3. user should be able to edit and delete the item from the show list
2. Python data: read a dataset in excel format and remove unwanted data, calculate MAX, MIN, AVG for each item after cleaning up, export the result in excel format
 - 2.1. the application should be able to get the file from the user and read it

2.2 application should be able to get the column and the operation that the user wants to analyze

2.3 application should save the result and input information for each set of data

3. Web and Mobile: Coffee ordering application

3.1 application should be able to get the coffee type and user-selected ingredient and its final price

3.2 application should be able to show the list of user ordered items (order history)

3.3 application should allow users to update the list item or remove them

Lesson 1: My Favourite Language (50 minutes)

Time	Learning Activity	Assessment	Material	Resources/ Technology
15	<p>Going through Roadmap to discuss the basics of each Software development path to cover the most anticipated/modern languages in the programming world</p> <p>NOTE: If any of the students have a prior professional knowledge ask him to use the same language and assist you in the teaching process by helping a student that has a less programming knowledge</p>	<ol style="list-style-type: none"> Students will find their language of choice and put their reasons and thoughts in their e-file (Portfolio/CV/Google Doc/etc.). Student Draw their road map on their e-file 	Roadmap	<p>Students can search their chosen language at Udemy or Pluralsight</p> <p>Google Doc Google Draw Microsoft Word Microsoft Powerpoint</p>
15	<ol style="list-style-type: none"> Discuss with students about their chosen path Students can discuss together, and also the teacher can help those with confusion. 	Submit their chosen language to the discussion board,		<p>Using provided application to submit the final decision</p> <p>YoTeach</p>
10.	Break Time (Motivate students to get around and talk (Discuss) about their chosen language and their previous possible knowledge and ability in each language)			
10	<ol style="list-style-type: none"> Group students based on their chosen language (5 or less group) Students can do the rest of the units as peers (Max 3 people in a group). 			

Lesson 1: Homework

Homework	Resources	Result
Use a free online learning platform to learn about their language of choice	Khan Academy FreeCodeCamp Student Hub.	<p>They would be ready to do the more complex calculation on user input and create more human-readable results.</p> <p>They should know about</p> <ol style="list-style-type: none"> array (advance). map (loop).

		<ol style="list-style-type: none">do/while.functions/method.class.
--	--	--

Lesson 2: Know your language (50 minutes)

Time	Learning Activity	Assessment	Material	Resources/ Technology
30	Download and set up the best IDE Note: <ol style="list-style-type: none"> 1. Java: JetBrains 2. Swift (needs mac): XCode 3. Kotlin: Android studio 4. Python: PyCharm 5. Javascript: Visual Studio Code 			
10	Break Time (Help students who have problems with IDE setup)			
20	Looking for their language of choice and finding software that is related to their path in development <ol style="list-style-type: none"> 1. The teacher will help them to find the best possible one for each group of language 2. The teacher should have a list of boilerplate based on best practices in top languages such as C#, JAVA, JS, KOTLIN, SWIFT, PYTHON Note: Desktop/Android/iOS/C++ have their own boilerplate in their IDE Javascript: NodeJS Python: CookieCutter	Students will create an account in GitHub.com with their student id Add the boilerplate to their GitHub account using CLI or GUI Applications	GitHub.com	Using a source control software like Git download a boilerplate and start observing and also as a starting point They usually can search for boilerplate in their language, and there should be plenty of codes What Is GitHub? A Beginner's Introduction to GitHub

Lesson 2: Homework

Homework	Resources	Result
Play around with their IDE and start getting familiar with their feature and plugins that could help them to	Their IDE website and Youtube (Students are encouraged to locate the video)	Learning shortcuts and get comfortable using the IDE

Lesson 3: Having fun with your language - Hanooy Towers (50 minutes)

Time	Learning Activity	Assessment	Material	Resources/ Technology
20	<p>Start with a simple algorithmic problem like Hanooy Tower.</p> <p>Break the problem into small pieces. Detect the pattern in the solution. Create the Flowchart for the problem</p>		Miro	Top 25 Algorithms
10	Break Time (Help students who have difficulty with IDE setup)			
30	<p>Start working on boilerplate from GitHub and add essential functionality to the already written structure</p> <ol style="list-style-type: none">1. Only one function could be enough2. The function should include the structures like array3. The function should include condition flow for loop and recursion.	<p>Run their function and get the result.</p> <p>To push the result and updated code in their git.</p>		

Lesson 3: Know libraries in your language

Time	Learning Activity	Assessment	Material	Resources/ Technology
30	<p>Ask the students to find packages and libraries related to each of the below items in their language</p> <ol style="list-style-type: none"> 1. Reading/Writing files 2. Querying data in file 3. Get input from the end-user 4. Print/send data after processing to the end-user. <p>Ask students to talk about their ideas for a new application.</p> <p>If their idea covers all four subjects, they can implement it instead of the prepared problem</p>		IDE Computer	GitHub.com JSON
10	Break Time (Help students who have issues with IDE setup)			
20	<p>Read a file (provided) including a list of items and prices. Their category</p> <p>Return the total prices for each type</p>			

Lesson 3: Homework

Homework	Resources	Result
Reading the documentation for the related package for each to be more advanced and comfortable to use them in a real-world project	The Packages websites.	Get prepared for their next step.
Home assessment	<p>Create one of the</p> <ol style="list-style-type: none"> 1. a blog post 2. a video <p>About why they have chosen the language to work with. Finally, post</p>	<p>Use Medium or dev.to for blog posts.</p> <p>Use Flipgrid or Youtube for the video, TikTok or any Video-based social media.</p>

	<p>it on their portfolio (Ask them to be as ambitious as possible about their future in programming)</p> <p>What they want to do/and what they are interested in).</p> <p>Students with their Idea also can talk about their idea in their presentations</p> <p>*Rubric 1</p>	<p>Use Google Drow or Slide Or any other software they are comfortable with for Drawing</p> <p>Using Social media like Medium dev.to Flipgrid TikTok</p>
--	---	--

Lesson 4: Start working on the application (50 minutes)

Time	Learning Activity	Assessment	Material	Resources/ Technology
20	<p>Break down the application into three major stages. Find the packages that are needed for each stage.</p> <p>Install packages, and get ready for the code.</p> <p>Each person in the group should pick one stage, break it down into small steps, and create a flowchart for the stage.</p> <ol style="list-style-type: none"> 1. Getting data from the user input and adding them to a file. 2. Reading the file and process data to show them to user 3. Edit the file and add data to existing data. 	<p>Submit the final flowchart using Miro</p>	Miro	Tips for better flowchart
10	Break Time (Help students who have problems).			
30	<p>Start coding the first step. Create a file based on the structure needed to store data. Add some sample data to file</p> <ol style="list-style-type: none"> 1. Get the input from users 2. Keeping them to file <p>File should be in text, CSV, or JSON format based on their chosen language</p> <p>Simple GUI to get input data</p>	<p>Student should push their code to GitHub</p> <p>*Rubric 2</p>	IDE, A Computer	<p>Student can start their pair programming using screen sharing tools or code sharing tools</p> <p>VScode Microsoft Teams</p>

Lesson 4: Homework

Homework	Resources	Result
If anything remained from their stage one coding should be completed as homework (Email teacher or work with other groups to finish the tasks)		Completed stage one

Lesson 5: Get your application to the next level (50 minutes)

Time	Learning Activity	Assessment	Material	Resources/ Technology
30	<p>Start the second stage of the application.</p> <ol style="list-style-type: none"> 1. Read the created file 2. Show data after processing the file <p>Simple GUI to show the output data</p>	<p>The student should push their code to GitHub</p> <p>*Rubric 2</p>	<p>Their chosen IDE and given JSON file</p>	<p>JSON</p>
10	Break Time (Help students who have problems with flowcharting)			
20	<p>Students should add both code documentation and flow documentation to their work using the standard format for each language in code documentation.</p> <p>Also, get familiar with MD files.</p>	<p>Students should add Readme.md file to their project</p> <p>*Rubric 3</p>	<p>Make a Readme</p> <p>Learn the basics of writing an md file</p>	<p>MD file editor Dillinger</p>

Lesson 5: Homework

Homework	Resources	Result
<p>If anything remains from their stage two coding should be completed as homework (Email teacher or work with other groups to finish the tasks)</p>		<p>Completed stage two</p>

Lesson 6: Real-world problem needs a real-world solution (50 minutes)

Time	Learning Activity	Assessment	Material	Resources/ Technology
30	<p>Start the third stage of the application.</p> <ol style="list-style-type: none"> 1. Read the created file 2. Get user input data 3. Update the file with new input data. <p>Simple GUI to update data</p>	<p>Students should push their code to GitHub</p> <p>*Rubric 2</p>	IDE Computer	No new resource
10	Break Time (Help students)			
20	<p>Complete their work in stage three (if any remained).</p> <p>Update both code documentation and flow documentation to their work</p>	<p>Students should add Readme.md file to their project</p> <p>*Rubric 3</p>	IDE Computer	No new resource

Final Assessment: Homework

Assessment	Resources	Result
<p>Create a Video tutorial of their work.</p> <p>Explain their language.</p> <p>Explain the Stages of coding.</p> <p>Explain the flowchart of each stage.</p> <p>Show the documentation</p>	<p>*Rubric 4</p> <p>Note: Students with social anxiety can write another article to explain their flow</p>	<p>Final Grade</p> <p>20 percent by their teammate</p> <p>20 percent by other groups</p> <p>60 percent by educator</p>

Rubrics

Video presentation rubric (20 minutes long) *Rubric 1 and *Rubric 2

Criteria	1 point	2 points	3 points	4 points
Organization	The presentation was difficult to follow due to the disorganization of the utterances.	The presentation was not easy to follow.	The presentation was easy to follow.	The presentation was very easy to follow.
Accuracy of language uses	It was hard to understand due to incorrect use of vocabulary and grammar.	The student made some mistakes in vocabulary and grammar.	The student made a few mistakes in vocabulary and grammar but there were no patterns of errors.	The student communicated well using correct vocabulary and grammar.
Understanding of topic	The student did not show an adequate understanding of the topic.	The student clearly understood most aspects of the topic and presented their information with ease.	The student seemed to understand the main points of the topic and presented those with ease.	The student clearly understood the topic in-depth and presented his/her information convincingly.
Accuracy of information	Most of the information was inaccurate or not clear.	Most information presented in writing was clear but was not usually accurate.	Most information presented in writing was clear, accurate and thorough.	All information presented in the video was clear, accurate and thorough.
How close they follow the code in their presentation	Not all important parts could be found in the video and also they missed the explanation of non-technical topics.	None of the non-technical topics were covered in the video, and also, not all code information was presented.	They cover the most important topics in their code professionally but could not cover all non-technical topics.	They covered all the topics and codes in their project, including git and IDE.

Writing Article Rubric (Less than 5000 words) *Rubric 1 and *Rubric 3

Criteria	1 point	2 points	3 points	4 points
Organization	The article was challenging due to the disorganization of the utterances.	The article was not easy to follow.	The article was easy to follow.	The article was straightforward to follow.
Accuracy of language uses	It was hard to understand due to incorrect use of vocabulary and grammar.	The student made some mistakes in vocabulary and grammar.	The student made a few mistakes in vocabulary and grammar, but there were no patterns of errors.	The student communicated well using correct language and grammar.
Understanding of topic	The student did not show adequate knowledge of the topic.	The student clearly understood most aspects of the topic and wrote their information easily.	The student seemed to understand the main points of the topic and presented those with ease.	The student clearly understood the topic in-depth and wrote their information convincingly.
Accuracy of information	Most of the data was inaccurate or not explicit.	Most information written in writing was clear but was not usually accurate.	Most information written in writing was clear, precise and thorough.	All information written in writing was clear, valid and rigorous.
How close did they follow the code in their presentation,	Not all essential parts could be found in the article, and they missed the explanation of non-technical topics.	None of the non-technical topics were covered in the report, and also, not all code information was presented.	They professionally cover the most important topics in their code but cannot cover all non-technical topics.	They covered all the topics and codes in their project, including git and IDE.

Code Quality *Rubric 2

Criteria	1 point	2 points	3 points	4 points
Syntax	Program does not compile or (in a dynamic language) contains typographical errors leading to undefined names.	Program compiles but contains errors that signal misunderstanding of syntax such as the semicolon in <code>if(exp);{}</code>	Program compiles and is free from major syntactic issues, breadableontain non-standard usage or superfluous elements.	Program compiles and contains no evidence of misunderstanding or misinterpreting the syntax of the language.
Logic	Program contains some conditions that specify the opposite of what is required (less than vs. greater than), confuse Boolean AND/OR operators, or lead to infinite loops.	Program logic is on the right track with no infinite loops but shows no recognition of boundary conditions (such as <code><</code> vs. <code><=</code>).	Program logic is mostly correct but may contain an occasional boundary error or redundant or contradictory condition.	Program logic is right, with no known boundary errors and no redundant or contradictory conditions.
Correctness	Program does not produce correct answers or appropriate results for most inputs.	The Program approaches correct answers or appropriate results for most inputs but can contain miscalculations in some cases.	The program produces correct answers or appropriate results for most inputs.	The program produces correct answers or appropriate results for all inputs tested.
Clarity	The Program contains no documentation or grossly misleading indentation.	The Program contains some documentation (at least the student's name and program's purpose) but has occasionally misleading indentation.	The program contains some documentation on major functions, variables, or non-trivial algorithms. Indentation and other formatting are appropriate.	The program contains appropriate documentation for all major functions, variables, or non-trivial algorithms. Formatting, indentation, and other white space aid readability.

Modularity	Program is one big function or is decomposed in ways that make little sense.	The Program is decomposed into units of appropriate size, but they lack coherence or reusability. The program contains unnecessary repetition.	The program is decomposed into coherent units but may still contain some unnecessary repetition.	The Program is decomposed into coherent and reusable units, and unnecessary repetition has been eliminated.
------------	--	--	--	---

For Students with disability and limitations

- 1. The students with hearing difficulty**
can communicate through an IM messaging application with their group members
- 2. The students with social anxiety**
have support for different types of assessments
- 3. The students who have privacy concerns**
using social media to share their articles or videos can send the result directly to the teacher's email.
- 4. The students with physical disabilities**
will support with Voice to Code Tools and support from their teammates in writing the code if they want
- 5. The students with an IEP (Individual Education Plan)**
will have their program and work based on the defined plan
- 6. The students with ADHD**
will have direct support and a limited group from the teacher.

The original Unit Plan

Link:

<https://www2.gov.bc.ca/assets/gov/education/kindergarten-to-grade-12/teach/pdfs/curriculum/appliedskills/2003infotech1112.pdf>

Title:

Information and communications technology 11 and 12

Unit:

Arrays, Files, and Searching (p. 82/140)

It is expected that students will:

- implement sequential and binary searching techniques
- select a conventional variable or array based on the type and use of data to be stored
- create programs that apply the use of file techniques
- use one-dimensional and parallel array indexing, naming, and sizing techniques
- create programs that use and manipulate array structures to solve problems
- apply structured programming techniques to solve complex problems

Suggested instructional strategies:

Students develop proficiency in programming to solve complex problems.

- Brainstorm with students the types of data storage that require arrays. Have groups of students develop array structures that could be used to store specific data types.
- Ask students to generate sample data for a data structure and enter it into a text file. Then ask them to use the programming language to open the file and read the values into the data structure.
- Have small groups of students modify data fields and add new data to an existing program.
- Have students complete a project using arrays to store data. Ask them to: - reverse data in an array - do palindrome testing - and parse text data for correct spelling and spacing
- Invite students to work in teams to solve a problem that involves the use of arrays. Give them 10 or 15 minutes to work independently on a solution. Then ask each student to move to the workstation of another student in their team and continue programming the solution. Repeat this process until every team member has worked at each of the team workstations.
- Have students maintain a procedure or program fragment library with code that they can reuse on future assignments. This library should be updated throughout the course.

References

Ahmed, K. (n.d.). *Developer Roadmaps*. Roadmap.sh. Retrieved April 22, 2022, from <https://roadmap.sh>

Android Studio. (2019). *Download Android Studio and SDK tools*. Android Developers. <https://developer.android.com/studio>

CSD Guide to Assessment. (n.d.). Google Docs. Retrieved April 22, 2022, from <https://docs.google.com/document/d/1CQiGBXndrfKnK0Yro0vF8o01UBberzGe0eV1obvoCc0/edit>

Flowchart Tips - Five Tips for Better Flowcharts. (n.d.). Wwww.smartdraw.com. <https://www.smartdraw.com/flowchart/flowchart-tips.htm>

Freak, C. (2021, March 6). Top 25 Algorithms Every Programmer Should Know. Techie Delight. <https://medium.com/techie-delight/top-25-algorithms-every-programmer-should-know-373246b4881b>

freeCodeCamp.org. (n.d.). Wwww.freecodecamp.org. <https://www.freecodecamp.org>

Use Microsoft Live Share to collaborate with Visual Studio Code. (n.d.). Code.visualstudio.com. Retrieved April 22, 2022, from <https://code.visualstudio.com/learn/collaboration/live-share>

Microsoft. (2016, April 14). Visual Studio Code. Visualstudio.com; Microsoft. <https://code.visualstudio.com>

PyCharm: the Python IDE for Professional Developers by JetBrains. (n.d.). JetBrains. <https://www.jetbrains.com/pycharm>

Miro | Free Online Collaborative Whiteboard Platform. (n.d.). Https://Miro.com/. <https://miro.com>

Make a README. (n.d.). Make a README. Retrieved April 22, 2022, from <https://www.makeareadme.com>

Online Markdown Editor - Dillinger, the Last Markdown Editor ever. (n.d.). Dillinger.io. Retrieved April 22, 2022, from <https://dillinger.io>

JSON. (n.d.). Wwww.json.org. <https://www.json.org/json-en.html>

Jackson, B. (2018, April 20). *What Is GitHub? A Beginner's Introduction to Git*. <https://kinsta.com/knowledgebase/what-is-github>

Khan Academy. (2010). Khan Academy. Khan Academy. <https://www.khanacademy.org>

Sadadow. (n.d.). *Student Hub Overview - Microsoft Learn Student Hub*. Docs.microsoft.com. Retrieved April 22, 2022, from <https://docs.microsoft.com/en-ca/learn/student-hub>

Mcdaniel, R. (2010, June 10). *Bloom's Taxonomy*. Vanderbilt University; Vanderbilt University. <https://cft.vanderbilt.edu/guides-sub-pages/blooms-taxonomy>

NFLRC | National Foreign Language Resource Center. (n.d.). Video Presentation Rubric, Retrieved April 22, 2022, from <https://nflrc.hawaii.edu>

PALMS Backchannel Chat | The new alternative to Todaysmeet. (n.d.). Yoteachapp.com. Retrieved April 22, 2022, from <https://yoteachapp.com>

Pluralsight - Unlimited Online Developer, IT, and Cyber Security Training. (n.d.). Wwww.pluralsight.com.
<https://pluralsight.com>

Online Courses - Learn Anything, On Your Schedule | Udemy. (2015). Online Courses - Learn Anything, On Your Schedule | Udemy. Udemy; Udemy. <https://www.udemy.com>