Crack the Code

Gr. 8-12 Activity Write Up

Actua | 111 Murray Street, Ottawa, ON K1N 5M5 | www.actua.ca | 613 234 4137

Youth · STEM · Innovation
Jeunesse · STIM · Innovation

Crack the Code

Terms of Use

Prior to using this activity or parts thereof, you agree and understand that:

- It is your responsibility to review all aspects of this document and the associated activity write ups, and ensure safety measures are in place for the protection of all involved parties.
- Any safety precautions contained in the “Safety Considerations” section of the write-ups are not intended as a complete list or to replace your own safety review process.
- Actua shall not be responsible or liable for any damage that may occur due to your use of this content.
- You may adapt the content for your program (remix, transform, and build upon the material), providing appropriate credit to Actua and indicating if changes were made. No sharing of content with third parties without written permission from Actua.

About Actua

Actua is Canada’s leading science, technology, engineering and mathematics (STEM) youth outreach network, representing a growing network of over 40 universities and colleges across the country. Each year 350,000 young Canadians in over 500 communities nationwide are inspired through hands-on educational workshops, camps and community outreach initiatives. Actua focuses on the engagement of underrepresented youth through specialized programs for Indigenous youth, girls and young women, at-risk youth and youth living in Northern and remote communities. For more information, please visit us online at www.actua.ca and on social media: Twitter, Facebook, Instagram and YouTube!

Crack the Code

Activity Summary

In this activity, participants will learn about cryptography and how to select devices and online services that protect their data through good encryption and decryption practices. Participants will also learn about how to create strong and memorable passwords.

This activity is part of a series in the cyber smart education suite which includes; Digital Citizenship and You, Being Online, Web Detective, Netiquette, Crack the Code and Secure the Network. Explore Actua's Cyber Smart Educator Handbook to learn how you can bring cyber smart education into your teaching context.

Developed by Actua, 2021.

Delivery Environment	Activity Duration	Intended Audience	Tech
In-Person	1 hour & 30 mins	Grades 8-12 *(Ages 14-18)*	Certain activities will require a laptop/tablet. With modifications, it is possible to run this entire lesson in pairs/groups. **Facilitators should have access to a laptop, projector, speakers, and a screen or blank wall to project onto.** • Projector • Speaker • Screen/Blank Wall • Laptops/Tablets

Learning Outcomes

Following this activity participants will:

- Understand the role of cryptography in privacy management.
- Use and share key strategies to create strong, unique passwords and other strategies to keep personal data secure.

TOOLSETS	SKILLSETS	MINDSETS
Knowledge, resources, and experiences • Cryptography • Encryption • Decryption • Strong, unique passwords	**Digital skills, STEM skills, & essential employability and life skills** • Being safe and responsible online • Numeracy • Critical thinking • Analysis • Problem solving	**Digital intelligence, community action, and computational thinking** • Privacy management • Logical thinking • Computer literacy

Logistics (Timing, Group Sizing, Materials)

Section Title	Est. Time	Group Size	Materials
Opening Hook	10 minutes	*Whole Group*	**Facilitators** • Encrypted Message *(Appendix C)* • Decryption Key *(Appendix C)*
Section 1: Decrypting the Secret	20 minutes	*Whole Group*	**Facilitators** • The Caesar cipher | Journey i... • Crack the Code Slide Deck

Section Title	Est. Time	Group Size	Materials
Codes			
Section 2: Password Creation	20 minutes	*Individual; Small Group*	**Facilitators** • #WorldPasswordDay TikTok • Most common passwords of the year 2020 • Teach Students About Intern... • Board & Board Marker **Per Small Group** • Paper & Writing Utensil • Laptop/Tablet • https://www.experte.com/password-check
Section 3: Password Verifier	30 minutes	*Whole Group; Individual*	**Per Participant** • Password Verifier Sample Code *(Appendix C)* • Laptop/Tablet (if using PyCharm, must be a laptop/computer)
Reflection & Debrief	10 minutes	*Whole Group*	**Facilitator** • Code Breakers Of Bletchley ...

Safety Considerations

Safety considerations have been provided below to support safety during this activity, however they are not necessarily comprehensive. It is important that you review the activity and your delivery environment to determine any additional safety considerations that you should be implementing for the delivery of these activities.

Emotional Safety

The goal of this Cyber Smart project is to equip participants with the tools and knowledge to understand online behaviours and make safe decisions.

- Facilitators should understand that participants have different lived experiences and prior knowledge about cyber safety, cyber security, and digital citizenship. This activity may involve or lead to discussions of sensitive topics, such as cyberbullying and other online risks. Facilitators should always keep the participants' emotional safety in mind in these discussions, and defer to training from their institution and training received for this project.
- Facilitators should focus on guiding discussions toward an appreciation for healthy and safe online behaviours, and empowering participants to make responsible, informed and smart choices.

Online Safety

Some components of this activity require the use of devices connected to the internet.

- Facilitators should review the provided videos and read/explore provided websites and materials to determine if they are suitable for your participants.
- Where applicable, facilitators should remind participants to stay on task and only use links provided within this activity.
- Facilitators should also model and encourage appropriate online behaviour by all participants in the group (e.g., using chat boxes to answer and ask questions, using positive and encouraging language, using devices for the purpose of the task).

Curriculum Links

Each of these activities align with these components found in the Pan-Canadian K-12 Computer Science Education Framework:

Cyber Security

- Starting learners should be able to define cybersecurity and create safe passwords using effective criteria. Proficient learners should be able to describe common cyber attacks and identify malicious content, apply prevention practices and assess the role that people play in creating, preventing, and minimizing the impacts of cyberattacks as well as consider how they affect people and society (p. 24).

Data: Data Governance

- Starting learners should be able to identify ways that their digital or physical activity creates digital data and learn how to adjust privacy settings on commonly used digital tools. Proficient learners should be able to discover who owns the digital data they produce, as well as assess provincial, national and Indigenous data governance laws/agreements and be able to advocate for their data rights and the rights of others (p. 26).

Technology and Society: Ethics, Safety & the Law

- Starting learners should be able to identify strategies to protect their personal data and identity online. Proficient learners should be able to define and apply basic copywriter principles, explain privacy concerns, and assess the effects of computer crime/hacking on self and society (p. 28).

Activity Procedure

To Do in Advance

Section	Preparation

General	• **Think ahead and be ready to adapt:**
	  ◦ Determine your **delivery method** and leverage ideas from the delivery recommendations and adaptations sections.
	  ◦ While **estimated times** are provided, it will be helpful to think about how much time you would like to spend on different activities and discussions.
	  ◦ While **group sizes** (individual, pairs, groups) are suggested, many activities are flexible for whatever will work in your classroom.
	• **Prepare for the content:**
	  ◦ Have **answers in mind** to share with participants for the various reflection questions asked.
	  ◦ Examine the provided videos and read/explore the provided materials in *Appendix C* to determine if they are **suitable** for your participants.
	• **Equipment:**
	  ◦ Ensure device, screen and projector are set up.
	  ◦ Prepare participant devices.

Opening Hook	• Prepare the encoded message so that participants notice it as they walk into the room (ensure that the decryption key is hidden until needed in the Opening Hook, Step 2).
Section 1: Decrypting the Secret Codes	• Familiarize yourself with encrypting and decrypting. Try out each of the challenges: Crack the Code Slide Deck.

Section	Preparation
Section 2: Password Creation	• Review the website: Most common passwords of the year 2020 - scroll through the list to determine what is appropriate to share with participants before displaying.

Section 3: Password Verifier	• Familiarize yourself with the activity and code *Password Verifier Sample Code (Appendix C)* and Password Verifier Sample Code - Facilitator Support *(Appendix C)*
	• Determine a suitable Python IDE (integrated development environment for coding in Python) for your class. Examples include:
	  ○ PyCharm: free, requires download. If devices do not already have PyCharm, download beforehand.
	  ○ Repl.it: free, web-based browser that requires an account. Ensure this is suitable for your classroom before moving forward.

Opening Hook

10	13	14	22	11	4	3	6	4

8	18

15	14	22	4	17

1. Display the *Encrypted Message (Appendix C)* – write it on the board or project it on the screen). Give participants the challenge of determining who can decrypt this code in 1 minute before giving them the decryption key (step 2).
    a. Answer: Knowledge is power.
    b. ***Note:*** Participants may not be able to figure out how to decipher the message without the decryption key, but encourage them to try and let them know the decryption key will come soon.

2. Display the *Decryption Key (Appendix C)* (a number substitution cipher) and give them 2-3 minutes to decipher the message. ***Note:*** A is 0 and not 1.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
13	14	15	16	17	18	19	20	21	22	23	24	25

3. Connect what they just did to the process of encryption and decryption (see *Background Information in Appendix B* for more information).
    a. The big idea is that encryption is the process through which data is encoded so that it remains hidden from or inaccessible to unauthorized users. It helps protect private information, sensitive data, and can enhance the security of communication in apps and servers.

Section 1: Decrypting the Secret Codes

1. "What kind of information would you want to keep a secret (keep private) on the internet?"
    a. *Possible responses: passwords (e.g., device lock screen, social media, gaming), family photos, conversations with friends, school, work, location of your vacation home, where you like to shop, your health conditions, etc.*
        i. Any kind of personal information, especially data that identifies you, should be kept private - name, age, email address, phone number, photographs of you.
2. Let participants know they will be decrypting messages with the Caesar Cipher. Play this video for participants:

    The Caesar cipher | Journey into cryptography | Computer Science | Khan ...

    (Khan Academy, 0:00-**1:03s**)
3. Facilitate the Crack the Code Slide Deck (see Speaker Notes for more information). Give participants a paper and pencil for Challenge 1 - if they need something more challenging, look into the next challenges.

a. Participants just played with the process of encryption and decryption. The idea here is that messages that are harder to decrypt are better for protecting private information and sensitive data. This is similar to how accounts with passwords that are created to be strong and unique are harder to hack into.

Section 2: Password Creation

The process of encryption (as done in the activity above) is a simple step that creates extra protection for your password while it sits in a server and zooms across the internet. It essentially scrambles your password so it's unreadable and/or unusable by hackers. Encryption is one form of protection, but having a strong password is equally important.

1. Play this #WorldPasswordDay TikTok (OnlineKyne, 0:30s) created by Kyne Santos to show the math behind why this is important.
2. Consider some of the questions below to use as discussion prompts:
    a. "Why do we have passwords?"
        i. *Possible responses:* to help protect our personal information, to keep others from accessing our stuff, to stop strangers from logging onto our games online, to protect our identity because it verifies who we are online, to keep people out of our systems (e.g., Xbox or home Wi-Fi).
        ii. ***Note*:** As a prompt for participants who might not know what passwords are/have no experience with passwords, you can make a connection to locks (e.g., locks for bikes or diaries) and why locks are useful. Connections to strong passwords can include; ensuring that you have a different key to each lock you use, or for combination locks, not always using the same code or having a lock with a longer combination.
    b. **"What kind of accounts do you have/might have in the future that need passwords?"**
        i. *Possible responses: school emails, email account, online bank accounts, gaming accounts, Facebook, Instagram, Snapchat,*

TikTok, devices (your phone and laptop/computer should also be password protected).

c. **"Why is it important to create strong, <u>unique</u> passwords or authentication features?"**
    i. E.g., authentication features: fingerprint reader, face ID, PIN login, pattern login.

3. Scroll through this site to show participants the Most common passwords of the year 2020 [*remember to review prior to sharing - refer to the section "to do in advance" above*]. Point out how many other users use this password and the amount of time it took to crack it.
    a. *<u>Possible responses:</u> it's the first line of defense against a hacker (unauthorized user) to your computer and personal information, the stronger your password, the more protected your information will be from hackers (and malicious software).*
    b. Let participants know that strong passwords are important, but they should also ALL BE DIFFERENT (aka unique).
        i. Why? Because of credential stuffing: the idea that people plug in the same username and password for all of their accounts (e.g., Disney+ launched and had a number of their accounts 'hacked' because people re-used the same credentials (email + password) as other accounts.
4. Start a brainstorm on the board with the answers to participants' responses to: **"What methods can be used to create strong passwords and keep our information secure?"** (see Background Information in *Appendix B* for more information).
    a. Play this video for participants:
    Teach Students About Internet Safety and Privacy (Google for Education, 4:00s). Ask if participants have other strategies they'd like to add to the list.
5. Put participants into small groups and provide each with a paper and pencil. Assign each a different word. Their goal is to turn that word into a strong

password. At the end of each round, have them compare what passwords they created and how to help their peers make even stronger passwords.

   a. Examples of words: Chocolate, basketball, cheese pizza, coding, cyber security
   b. For each password, they need to apply the following:
      i. Make it a phrase or a series of seemingly unrelated letters/words (the video shows an additional step of taking only the first letters of a phrase to make a "non-sense" set of letters.
         1. ***Note*:** Google suggests using the following tips to help create longer passwords that are easier to remember: a lyric from a song or poem; a meaningful quote from a movie or speech; a passage from a book; a series of words that are meaningful to you.
      ii. Add in uppercase letters
      iii. Add in numbers
      iv. Add in a symbol
   c. Here's an example:
      i. **treasure (*Note*: This is the weak, starting password)**
      ii. Treasure **chest filled with gold** (+ make it into a phrase)
      iii. **T**reasure**C**hestfilledwith**G**old (+ strategically assign uppercase letters)
      iv. TCfwG (+ create an abbreviation - only keep the first letter of every word)
      v. TCfwG**01**d (+ add number(s) using a method that you will remember)
      vi. TCfw*****G01d***** (+ add symbols(s) using a method that you will remember)

6. Let participants test their made up passwords on https://www.experte.com/password-check. When you enter the password, it will show you how long it will take to crack! The longer the amount of time, the better. Have them try to recall their passwords from memory - this is helpful as

a reminder that making them long and complicated is important, but making them MEMORABLE is just as important.

Section 3: Password Verifier

Participants will analyze a Python program to validate whether an inputted password follows the conditions needed for a strong password.

1. Display the following code *Password Verifier Sample Code (Appendix C)* for the whole class.
    a. ***Note:*** Use the *Password Verifier Sample Code - Facilitator Support (Appendix C)* to help facilitate discussion and further understand the sample code if needed.
2. Depending on participants' familiarity with Python:
    a. Guide them through this code step by step,
        i. What words seem familiar?
        ii. What are some repeating worlds/symbols?
        iii. What do you think the conditions for the password to be valid are?
    b. Put participants into pairs/small groups to try and deconstruct and understand the code on their own.
    c. Have them develop their own Python program.
3. Have participants open up a Python IDE of your choosing (and IDE is an integrated development environment where participants can code in Python), examples include PyCharm (free, requires download) or repl.it (free, browser-based, account required). Participants will copy and paste the code into the IDE to test passwords. Have them come up with random passwords using the strategies from the previous exercise. If they are using the provided code, in order for the passwords to be valid, they must follow these conditions:
    a. At least 1 letter between [a-z] and 1 letter between [A-Z].
    b. At least 1 number between [0-9].
    c. At least 1 character from [$#@].
    d. Minimum length 6 characters.
    e. Maximum length 16 characters.

Reflection & Debrief

1. **“What are important reminders for ourselves, other participants and our families to keep passwords secure?”**
    a. *Possible responses: have different passwords for all accounts, log off of websites when you're using it, never share your password or write it down, use a password keeper tool, etc.*
2. **Career Connection:** Play video Code Breakers Of Bletchley Park (CBS, *0:00-**1:09s**). This video highlights the critical role women played during World War 2 as well as in this field in general.
3. Encourage participants to be a Cyber Smart Ambassador and share their learnings from this activity with their friends and family.

Delivery Recommendations

How might you deliver this content in different settings? Every activity has been designed for in-person delivery. Here, we provide recommendations for remote learning (online) or unplugged (no tech).

Remote (Online)	Unplugged (Low/No Tech)
General	
• Encourage participants to unmute themselves or type in the chat based on what is easiest for them to communicate. • Leverage a tool where participants can all participate online during discussions (e.g., Mentimeter, Jamboard, etc). • Make note of any links that need to be shared and be prepared to share them in the chat. • Use polls or other group interactions to check in and keep up engagement.	• Leverage boards to do brain storms/write down participant responses.
Opening Hook	
• Enlarge the encoded message on the screen.	• Teach about encryption and decryption with Lemon Secret Writing: https://www.spymuseum.org/education-programs/educators/lesson-plans-activities/

Remote (Online)	Unplugged (Low/No Tech)
Section 1: Decrypting the Secret Codes	
• Use the laser pointer option when facilitating the slide deck. • Activity can be done as-is online. For brainstorming, consider doing a verbal discussion or use a collaborative tool (e.g., Jamboard, Google Doc, Mentimeter).	• Instead of using the slide deck, write out an example on the board. On a large paper pad, write out the challenges for participants (*see slide deck*).
Section 2: Password Creation	
• Use break out rooms; have participants do tasks individually and share their made up password phrases in the chat to see whose is the strongest.	• Do the "test your password" game: participants will take steps based on a statement called out about password protection: https://curriculum.code.org/csf-19/coursec2/#powerful-passwords4
Section 3: Password Verifier	
• Find an appropriate web-based IDE for participants to test the program in.	• Print out the code for participants to analyze: what words are familiar to them? Can they try to explain the code? (Python is often written like a story).
Reflection & Debrief	
• Activity can be done as-is online. For brainstorming, consider doing a verbal discussion or use a collaborative tool (e.g.,	• Activity can be done as-is unplugged.

Remote (Online)	Unplugged (Low/No Tech)
Jamboard, Google Doc, Mentimeter).	

Delivery Adaptations

How might you adapt the time, space, materials, group sizes, or instructions to make this activity more approachable or more challenging? **Modifications** are ways to make the activity more accessible, **extensions** are ways to make the activity last longer or more challenging.

Modifications

GENERAL

- Ensure captions are on during videos played.
- Provide computer mouses where laptops are in use.
- Use pairs/groups instead of having participants work individually.

OPENING HOOK

- Use a single word and follow the same rules. This is also a great opportunity to encrypt another phrase, joke and song lyric that you participants are familiar with.

SECTION 1: DECRYPTING THE SECRET CODES

- Only focus on challenges 1-2 in the slides.
- Provide participants with partial solutions to their given codes to help guide them.
- Provide participants with additional time to decipher their message.
- Have students work in groups and give each group a print out of the alphabet.

SECTION 3: PASSWORD VERIFIER

- Break down the code together with participants.
- Print out the code and have participants work in groups to analyze the code instead of individually.

Extensions

SECTION 1: DECRYPTING THE SECRET CODES

- Participants can use this "Simple Encryption" activity created for Hour of Code.

SECTION 2: PASSWORD CREATION

- Have participants evaluate their personal passwords using the Government of Canada's Evaluation.
    - As a follow up question, you can ask participants to share how they will improve their passwords after using this evaluation.
- Test your password game (call out statements about password protection and take steps based on each statement).
- Play the password cracking challenge on Nova Lab's Cyber Lab (participants will need to complete the first virus attack challenge before being able to choose the password cracking challenge which will discuss hacking using guessing and brute force).
    - ***Note*:** Works best with Google Chrome.

SECTION 3: PASSWORD VERIFIER

- Have participants add in more code to create more conditions for the password.
- If participants are knowledgeable in Python, have participants write their own program. W3 Schools - Python Tutorial can be used as a Python refresher.

REFLECTION & DEBRIEF

- Participants can create a Canva Poster to share strategies with their friends and families on what cyber smart steps we can take when interacting with

new users online. Share this link with them https://www.canva.com/posters/templates/campaign/. It will be helpful to explore Canva to get an idea of how to use this resource yourself.

- Quickly show them how to create a new project and the different editing features they can use. If helpful, choose a suitable Canva template rather than have them find one themselves/have them draw it out.
- Participants can draw their creations on paper rather than on Canva.
- If time permits, have participants share their work.

References & Gratitude

Binance Academy. (2020, December). *History of Cryptography*. https://academy.binance.com/en/articles/history-of-cryptography

Betterteam. (2019, May 16). *Mathematician Job Description*. https://www.betterteam.com/mathematician-job-description

CBS. (2008, December 8). *Code Breakers Of Bletchley Park* [Video file]. https://www.youtube.com/watch?v=2458QZmNxRY

Cisco. (n.d.) *What is Multi-Factor Authentication?* https://www.cisco.com/c/en/us/products/security/what-is-multi-factor-authentication.html

Cyber Security Jobs. (n.d.) *Cryptographer Jobs & Careers*. https://www.cybersecurityjobs.net/cryptographer-jobs/

Google. (n.d.) *Create a strong password & a more secure account*. https://support.google.com/accounts/answer/32040?hl=en#zippy=%2Cmake-your-password-longer-more-memorable

Google for Education. (2017, June 25). *Teach Students About Internet Safety and Privacy* [Video file]. https://www.youtube.com/watch?v=25G4tLVH1JE&t=2s

Google for Education. (2021, November 8). *Python Regular Expressions*. https://developers.google.com/edu/python/regular-expressions

Government of Canada. (2020, January 15). *How strong is your password? Five ways to evaluate*. https://www.getcybersafe.gc.ca/en/blogs/how-strong-your-password-five-ways-evaluate

Government of Canada. (2020, March 2). *Passphrases, passwords and PINs*. https://www.getcybersafe.gc.ca/en/secure-your-accounts/passphrases-passwords-and-pins

Khan Academy. (2012, March 27). *The Caesar cipher | Journey into cryptography | Computer Science | Khan Academy* [Video file]. https://www.youtube.com/watch?v=sMOZf4GN3oc

MinuteVideos. (2017, January 30). *Cryptography and privacy. An easy explanation on how to create a key for encryption* [Video file]. https://www.youtube.com/watch?v=MU1Scwxc_RU

National Institute of Standards and Technology. (2019, December 9). *Back to basics: Multi-factor authentication (MFA)*. https://www.nist.gov/itl/applied-cybersecurity/tig/back-basics-multi-factor-authentication

Nord. (2020). *Top 200 most common passwords of the year 2020*. https://nordpass.com/most-common-passwords-list/

OnlineKyne. (2021, May 6). *#WorldPasswordDay* [Video file]. https://www.tiktok.com/foryou?is_copy_url=1&is_from_webapp=v1&item_id=6959281467310624006#/@onlinekyne/video/6959281467310624006

Python. (2022, February 12). *re — Regular expression operations*. https://docs.python.org/3/library/re.html

The Canadian Encyclopedia. (2018, Jul 16). *Cree Code Talkers*. https://www.thecanadianencyclopedia.ca/en/article/cree-code-talkers

Tutorials Point. (n.d.). *Python Regular Expressions*. https://www.tutorialspoint.com/python/python_reg_expressions.htm

Tutorials Teacher. (n.d.). *Python - if, elif, else Conditions*. https://www.tutorialsteacher.com/python/python-if-elif

University of Ottawa. (2020). *The Importance of passwords*. https://it.uottawa.ca/security/identity-authentication-theft

w3resource, (2020, February 26). *Python Exercise: Check the validity of a password*. https://www.w3resource.com/python-exercises/python-conditional-exercise-15.php

W3schools. (n.d.). *Python Tutorial*. https://www.w3schools.com/python/default.asp

Appendices

Appendix A: Career & Mentor Connections

ROYAL CANADIAN MOUNTED POLICE: CYBERCRIME INTELLIGENCE ANALYST

- A cybercrime intelligence analyst specializes in cybercrime, and uses that knowledge to develop strategies to identify criminal trends and patterns. They use this information to design strategic intelligence products, and provide expert advice on complex criminal investigations.

CYBER SECURITY PROFESSIONAL (INFORMATION SECURITY PROFESSIONAL)

- A cyber security professional identifies threats and vulnerabilities in various systems and softwares. They apply their knowledge to design security measures and implement solutions to defend against cybercrime, such as hacking and malware. These measures come in the form of technology and organizational processes.

CYBER SECURITY ANALYST (INFORMATION SECURITY ANALYST)

- A cyber security analyst monitors a company's computer networks and systems. In order to further protect the company from threats and breaches, they plan and implement security measures.

CRYPTOGRAPHER

- A cryptographer develops security systems using algorithms and ciphers to encrypt data. They ensure that important and sensitive data (e.g., financial, personal, business, etc.) is safe from unwelcomed eyes.

CRYPTANALYST

- A cryptanalyst analyzes and decrypts information in cipher texts and encrypted data (think of them as the opposite role of a cryptographer).

MATHEMATICIAN

- A mathematician supports organizations by using mathematical theories and techniques (e.g., collecting, analyzing and presenting data) to solve practical problems in various fields, like engineering, science, business and government.

Appendix B: Background Information

CRYPTOGRAPHY

Cryptography is the process of taking text written in regular everyday language and converting it into a secure code that can only be unlocked by someone it is meant for. In the modern world there are many ways to do this, most of which involve complex mathematical formulas. Cryptography is concerned with the entire process of secure communications and is the larger system that includes the smaller subsystems of encryption and decryption.

Some Historical Uses of Cryptography

- Egypt (3,900 years ago): Symbol replacement (the most basic form of cryptography) appears in ancient Egyptian and Mesopotamian writings. The earliest known example of this was found in the tomb of an Egyptian noble.
- Ancient India & Greek City-State of Sparta (later periods of antiquity): cryptography widely used to protect important military information
- Romans: creation and use of the Caesar Cipher - the most advanced cryptography in the ancient world
- American Military (as late as WWII): Thomas Jefferson in the 1790s invented the cipher wheel, a concept so advanced it was leveraged for American military cryptography until as late as WWII. Code talkers were employed by the military (WWI & WWII) to create an encryption system using words and phrases from Indigenous languages. These Indigenous soldiers used their knowledge of their native languages to encrypt and decrypt sensitive information during wartime.

ENCRYPTION/DECRYPTION

Encryption for the purposes of this activity means to encode a message so that it is unreadable to everyone except for who it is meant for. Decryption is the action of actually decoding that message so that it is readable. There are many different types of encryption, most of which use complex mathematical formulas to protect data. In this project we will be looking at two simple ciphers as a method of encryption. To learn more about what encryption is visit: https://www.w3schools.in/cyber-security/modern-encryption/

Caesar Cipher

The Caesar Cipher, named after Julius Caesar, is the simplest form of encryption.

1. *Encrypting* a message involves taking a message along with a key and shifting the characters over right in the alphabet by the number indicated by the key.
2. To *decrypt* the message, you use the key again and shift the characters left instead.

For Example: Our message is the word “JIM”.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T

Now let’s say we want to encrypt that message with a key of 3. The characters in our code would shift right three spots like this:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T

The encrypted message is now “MLP”:

- “J” shifted three spots over right to “M”,
- “I” moves three spots to the right to “L”
- “M” shifts three spots to the right to “P”.

If instead you were given “MLP” with the Key: 3 then you would do the opposite and shift the characters three spots left in the alphabet. To learn more visit: Caesar Cipher.

GOOD PRACTICES FOR STRONG PASSWORDS

- **Computer Science Connection:** Something that complements the strength of passwords is the emergence of biometric data (face, fingerprint, voice and now DNA).
- Use a unique password for every account.
- The longer the password, the better (phrase based passwords).
- At least 8 characters long, including lowercase and uppercase letters, numbers and symbols.
- Do not use personal information (e.g., birthdates, name, address).
- Always log off/sign out when leaving a site, especially when using public Wi-Fi or shared devices.
- Avoid entering passwords when connected to unsecured Wi-Fi connections (e.g., coffee shop Wi-Fi).
- Never tell anyone your password or write your password down.
- Avoid allowing your internet browser to “remember your passwords”.
- Make sure nobody is watching when you type in your password.
- Enable 2-factor (or even better, multi-factor) authentication.

MULTI-FACTOR AUTHENTICATION

This is a security process that asks for more than one factor of authentication from: something you know, something you are, and something you have.

- Something you know could be a password or access code.
- Something you are could be a biometric like a fingerprint or voice print.

- Something you have could be a phone or token.

Some examples could be:

- When logging into an email on a new device, you will need to have the password (something you know) and you will receive a prompt on your phone with a special code (your phone is something you have).
- When entering a secure building you may need an access code (something you know) and a security badge (something you have).
- To open your device you may need your fingerprint (something you are) and your password (something you know).

Appendix C: Additional Resources

OPENING HOOK

Image(s)

- Encrypted Message *(see below)*
- Decryption Key *(see below)*

SECTION 1: DECRYPTING THE SECRET CODES

Activity Slide Deck(s)

- Crack the Code Slide Deck

Video(s)

- The Caesar cipher | Journey into cryptography | Computer Science | Khan ... (Khan Academy, 0:00-**1:03s**)

SECTION 2: PASSWORD CREATION

Video(s)

- #WorldPasswordDay TikTok
- Teach Students About Internet Safety and Privacy (Google for Education, 4:00s)

Website(s)

- Most common passwords of the year 2020
- https://www.experte.com/password-check

SECTION 3: PASSWORD VERIFIER

Activity Page(s)

- Password Verifier Sample Code *(see below)*
- Password Verifier Sample Code - Facilitator Support *(see below)*

REFLECTION & DEBRIEF

Video(s)

- Code Breakers Of Bletchley Park (CBS, *0:00-**1:09s**)

What does this message say?

10	13	14	22	11	4	3	6	4

8	18

15	14	22	4	17

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Password Verifier

Steps

1. **Analyze the code under the "Password Verifier Code" section**
    a. What words seem familiar?
    b. What are some repeating words?
    c. What symbols do you recognize?
    d. What do you think the conditions for the password to be valid are?
2. **Code is the language of a computer.** Just like there are many different human languages (e.g., Tagalog, Arabic, Swahili, French), there are many types of computer languages. Python is one of them.
    a. Code should read like a book. Can you try to explain what the code is saying based on your current knowledge? The flowchart below might be helpful.
3. **Copy and paste the code below into the Python IDE** (PyCharm, or another of your choosing). Make sure you copy every line of code (indicated by the bolded black text) or you will run into errors.

Password Verifier Code

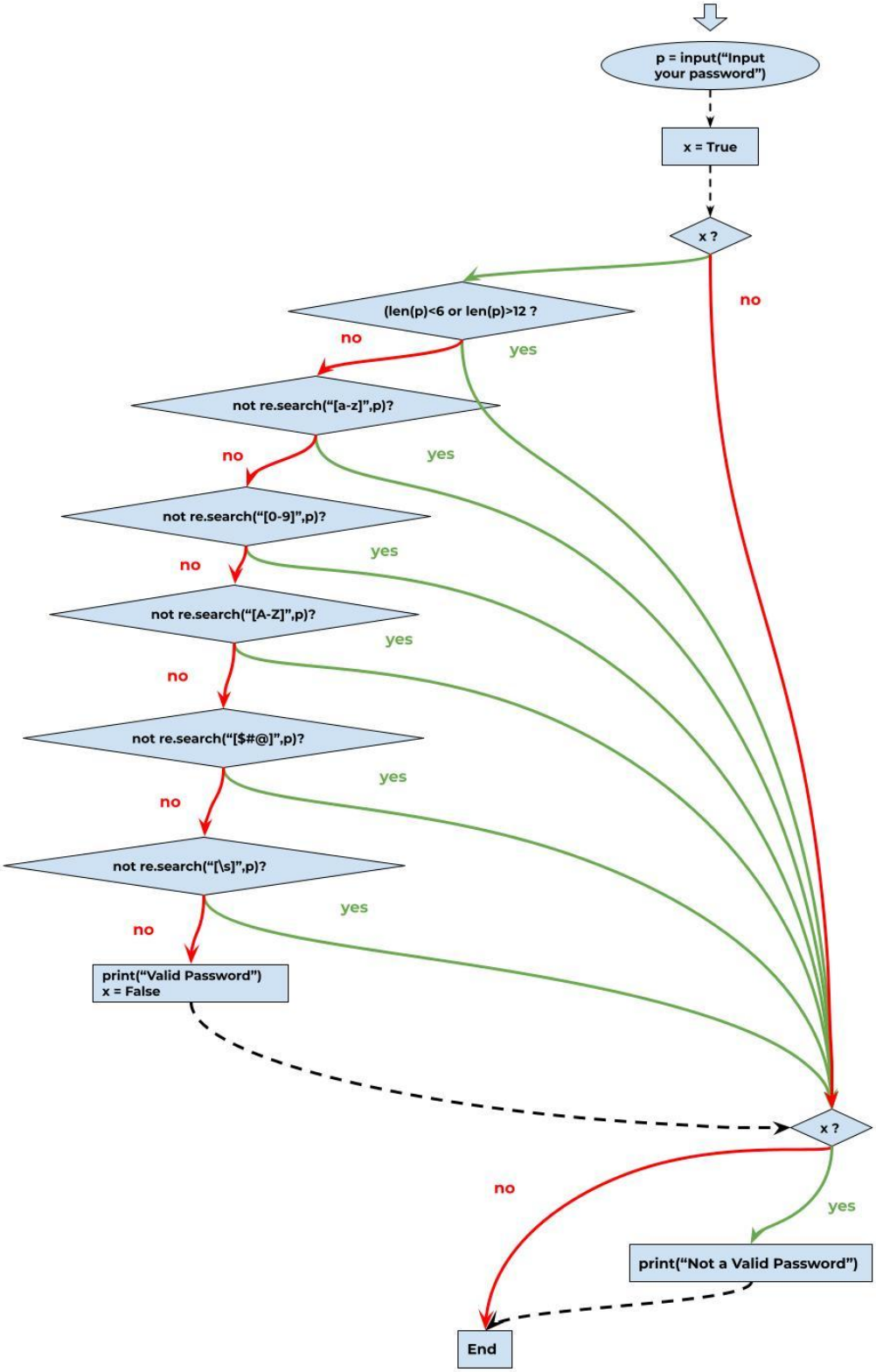
The following sample code is from w3resource "Python Exercise: Check the validity of a password". Copy the bolded code below ⬇️

```
# Python program to check the validity of a password (input from users)
import re
p= input("Input your password")
x = True
while x:
  if (len(p)<6 or len(p)>12):
     break
  elif not re.search("[a-z]",p):
     break
  elif not re.search("[0-9]",p):
     break
  elif not re.search("[A-Z]",p):
     break
  elif not re.search("[$#@]",p):
     break
  elif re.search("\s",p):
     break
  else:
     print("Valid Password")
     x=False
     break

if x:
  print("Not a Valid Password")
```

Flowchart

The following flowchart is adapted from w3resource "Python Exercise: Check the validity of a password).



actüa
Youth · STEM · Innovation
Jeunesse · STIM · Innovation

Password Verifier (Facilitator Support)

Answer Guide for Questions (Step 1)

1. **Analyze the code under the "Password Verifier Code" section**
    a. What words seem familiar? Point out English words they might be familiar with: import, input, true, if, false, print.
    b. What are some repeating words? Elif (an "else if statement")
    c. What symbols do you recognize? ( ) " " : [ ] ,
    d. What do you think the conditions for the password to be valid are?
        i. At least 1 letter between [a-z] and 1 letter between [A-Z].
        ii. At least 1 number between [0-9].
        iii. At least 1 character from [$#@].
        iv. Minimum length 6 characters.
        v. Maximum length 16 characters.

Break Down of Password Verifier Code

Line	Code	Explanation
Line 1	# Python program to check the validity of a password (input from users)	Any text that follows the hashtag symbol (#) on the same line is ignored by the Python interpreter. Here, we are using it to make a comment for ourselves.
Line 2	import re	"re" refers to **regular expressions**. The most common use of regular expressions is form validation, i.e. email validation, password validation, phone number

Line	Code	Explanation
		validation, and many other fields of the form. A regular expression (or RE) specifies a set of **strings** (string being a sequence of Unicode characters) that matches it; the functions in this module let you check if a particular string matches a given regular expression.
Line 3	p= input("Input your password")	The words “Input your password” will appear on the user’s device screen and they will be prompted to **input** and answer. In this case, if their input is true (meaning it meets all of the conditions set for a strong password), the computer will display “Valid Password”. If their input is false (meaning it does not meet all of the conditions set for a strong password), the computer will display “Not a Valid Password”. Point out the indentations. Python uses indentation to indicate a block of code. Python will give you an error if you skip the indentation. If their indents do not copy over, they can input them manually using the tab button on the keyboard.

Line	Code	Explanation
Line 4-5	x = True while x:	x will be True (and therefore get “Value Password) **while** it meets the following conditions...
Line 6-7	if (len(p)<6 or len(p)>12): break	Any **Boolean expression** evaluating to True or False appears after the **if** keyword. Python supports the usual logical conditions from mathematics: • Equals: a == b • Not Equals: a != b • Less than: a < b • Less than or equal to: a <= b • Greater than: a > b • Greater than or equal to: a >= b These conditions can be used in several ways, like in "if statements" in this example. Think length for the **len**() function. When the object is a string, the len() function returns the number of characters in the string. In this case, the computer is looking for a password between 6 to 12 characters long.
Line 8-9	elif not re.search("[a-z]",p): break	The **elif** keyword is Python's way of saying "if the previous conditions were not true, then try this condition". The **re.search**() method takes a regular

Line	Code	Explanation
		expression pattern and a string and searches for that pattern within the string. The code for this password verifier is saying that if the 6-12 character length password does not include a letter from a to z, the computer will display “Not a Valid Password” for the user to see.
Line 10-11	elif not re.search("[0-9]",p): break	Similar to [a-z] but for numbers now. In this example, the user must include a number. In addition to needing a length of 6-12 characters and at least one letter from a to z, the code for this password verifier is saying that if the inputted password does not include a number, the computer will display “Not a Valid Password” for the user to see.
Line 12-13	elif not re.search("[A-Z]",p): break	In addition to the above conditions, if the user does not include a capital letter from A to Z then “x=False” and the computer will display “Not a Valid Password”.
Line 14-15	elif not re.search("[$#@]",p): break	if the user does not include include one of these symbols: [$ # @] then “x=False” and the computer will display “Not a Valid Password”. Additional symbols can be added between the quotations if desired.

Line	Code	Explanation
Line 16-17	elif re.search("\s",p): break	For Unicode patterns, it will check if it can find a white space characters (e.g. spaces or tabs). This line is saying that in addition to the above conditions for the password, as long as it does not include a white space, then the computer will continue to read the line instead of displaying “Not a Valid Password”.
Line 18-19	else: print("Valid Password")	The **else** keyword catches anything which isn't caught by the preceding conditions. In python, the print statement: print(“TEXT”) is used to display text on the user’s device screen. In this case, if the above conditions are met, the computer will print “Valid Password”.
Line 20-24	x=False break if x: print("Not a Valid Password")	Notice the indentation is pushed back to the left. In this case, one or more of the conditions were not followed. If x=False, the computer will print “Not a Valid Password”.

```
# Python program to check the validity of a password (input from users)
import re
p= input("Input your password")
x = True
while x:
  if (len(p)<6 or len(p)>12):
     break
  elif not re.search("[a-z]",p):
     break
  elif not re.search("[0-9]",p):
     break
  elif not re.search("[A-Z]",p):
     break
  elif not re.search("[$#@]",p):
     break
  elif re.search("\s",p):
     break
  else:
     print("Valid Password")
     x=False
     break

if x:
  print("Not a Valid Password")
```

References & Gratitude

Google for Education. (November 8, 2021). *Python Regular Expressions*. Retrieved from https://developers.google.com/edu/python/regular-expressionshttps://developers.google.com/edu/python/regular-expressions

Python. (February 12, 2022). *re — Regular expression operations*. Retrieved from https://docs.python.org/3/library/re.html

Tutorials Teacher. (n.d.). *Python - if, elif, else Conditions*. Retrieved from https://www.tutorialsteacher.com/python/python-if-elif

Tutorials Point. (n.d.). *Python Regular Expressions*. Retrieved from https://www.tutorialspoint.com/python/python_reg_expressions.htm

w3resource. (February 26, 2022). *Python Exercise: Check the validity of a password.* Retrieved from https://www.w3resource.com/python-exercises/python-conditional-exercise-15.php

W3schools. (n.d.). *Python Tutorial*. Retrieved from https://www.w3schools.com/python/default.asp