Cryptography and Network Security

Documentation & Readme file for Assignment 1

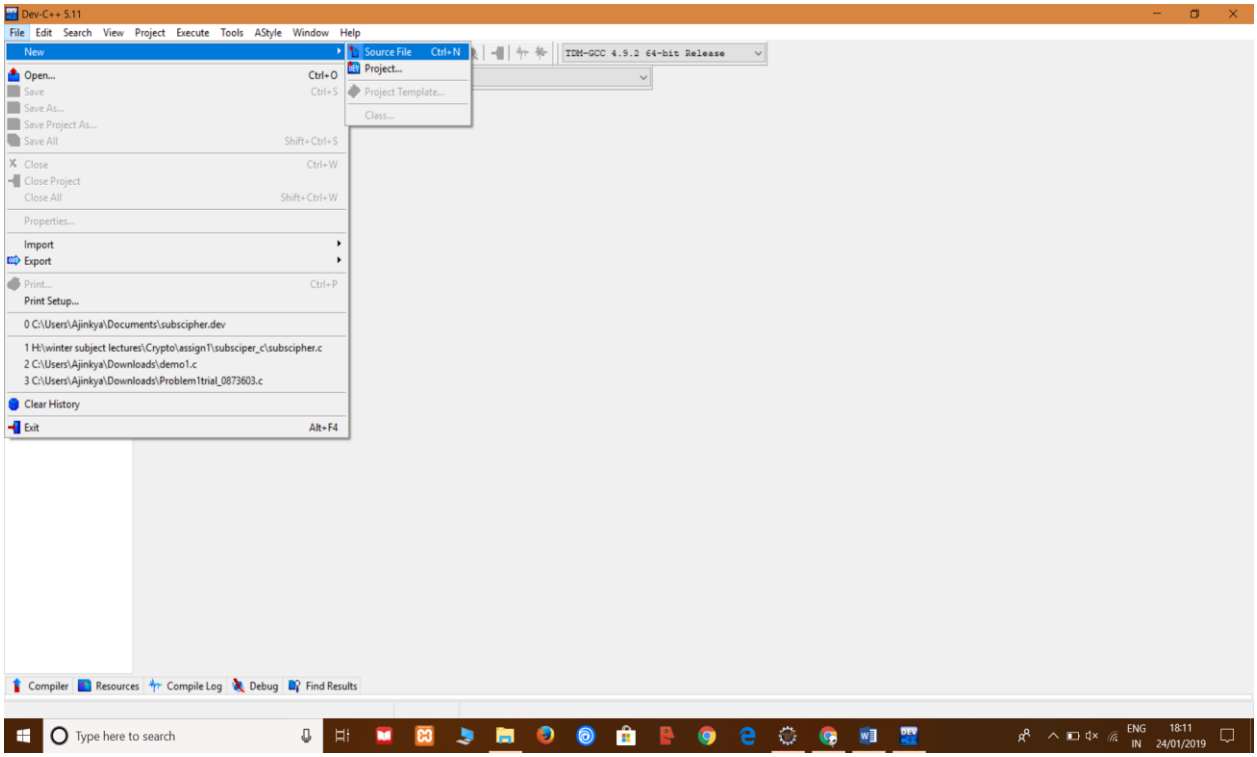
Problem 1

1. Write computer programs for the Substitution Cipher based on Z29 which is corresponding to 26 alphabetic characters (0 - 25), space (26), and “, ” (27) “.”(28). The key is a random permutation π on Z29. Write down encryption and decryption programs. Select a paragraph of text (I don’t think any two people will choose a same paragraph if they choose independently) and encrypt it using your encryption algorithm. Then use your decryption program to check the correctness. You can use Java, C or other computer languages. Record your plaintext, cipher text and the key π in your answer sheet

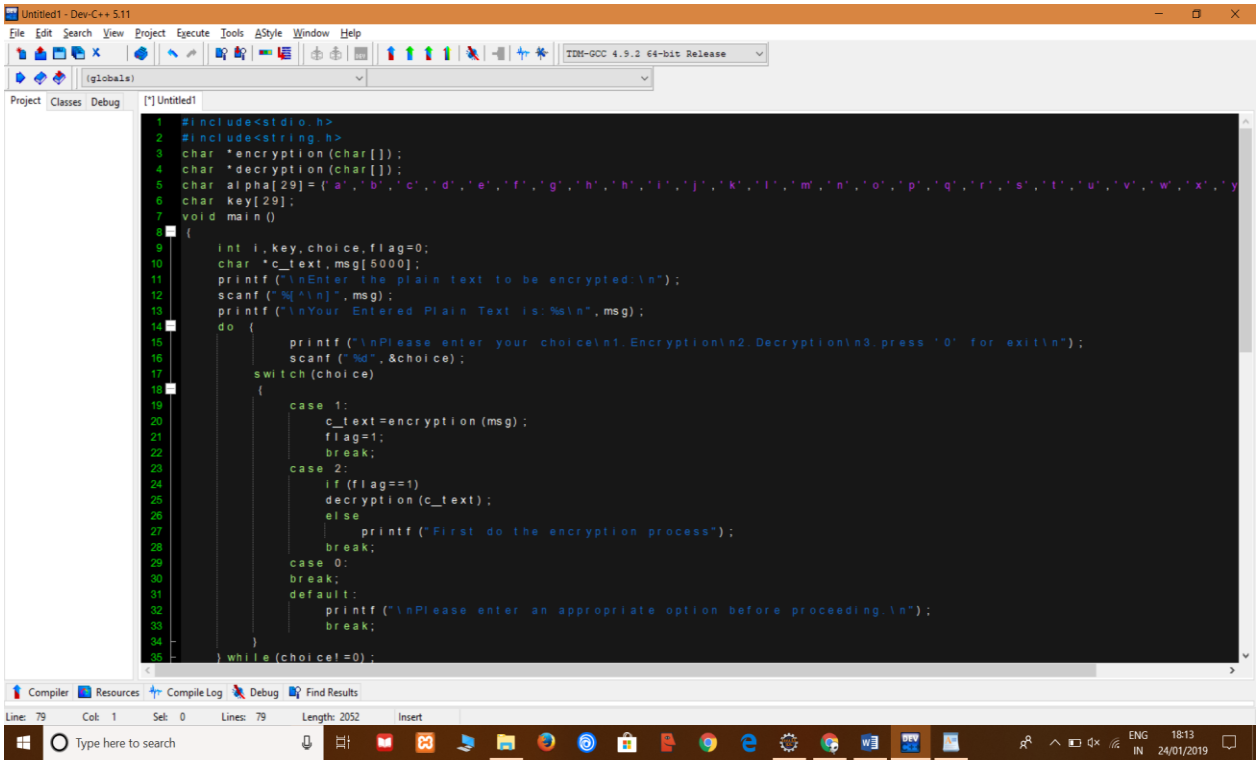
Solution: (Folder name: subsciper, file name: subsciper.c)

- Language of choice: C language
- Software/IDE used: Dev-C++
- You can download the software from - https://sourceforge.net/projects/orwelldevcpp/
- Name of the folder/file – subsciper/subscipher.c

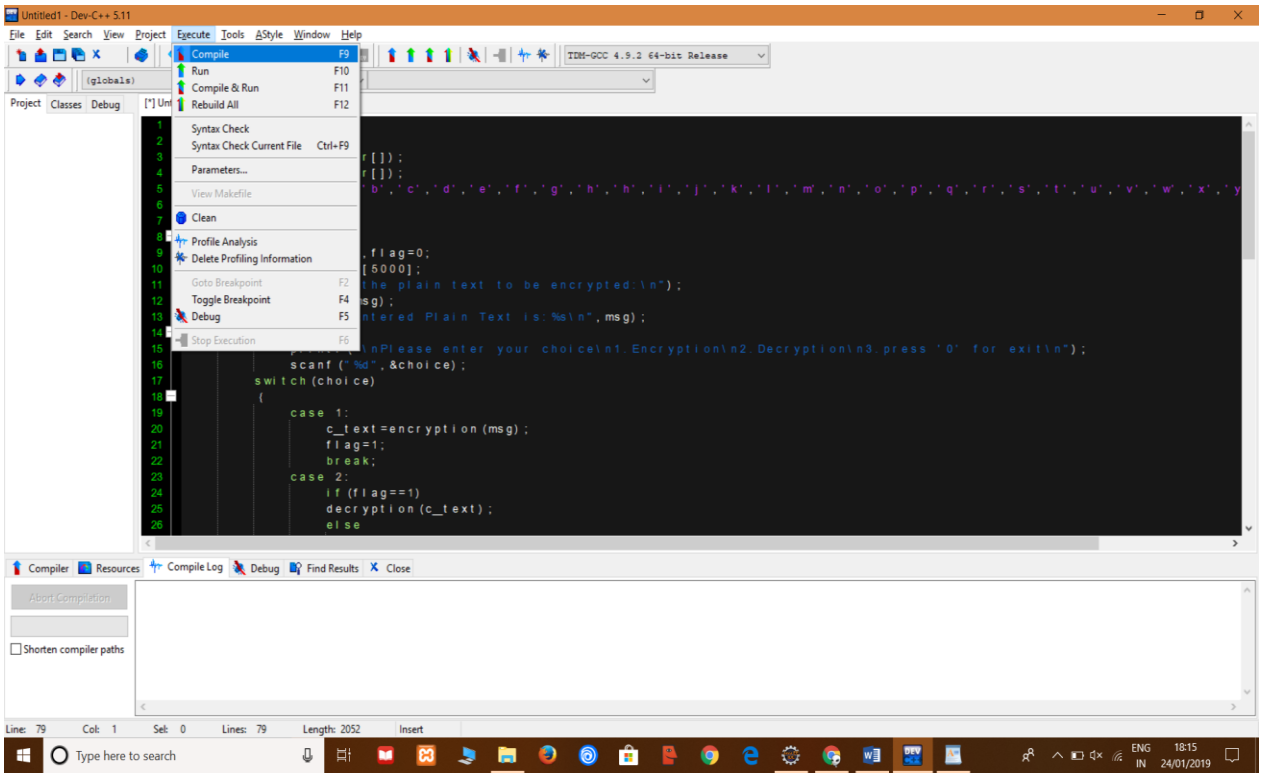
1. Download the software from above link and and create a new file as shown in below figure
2. Copy paste the contents of ‘subscipher.c’ into the new file created

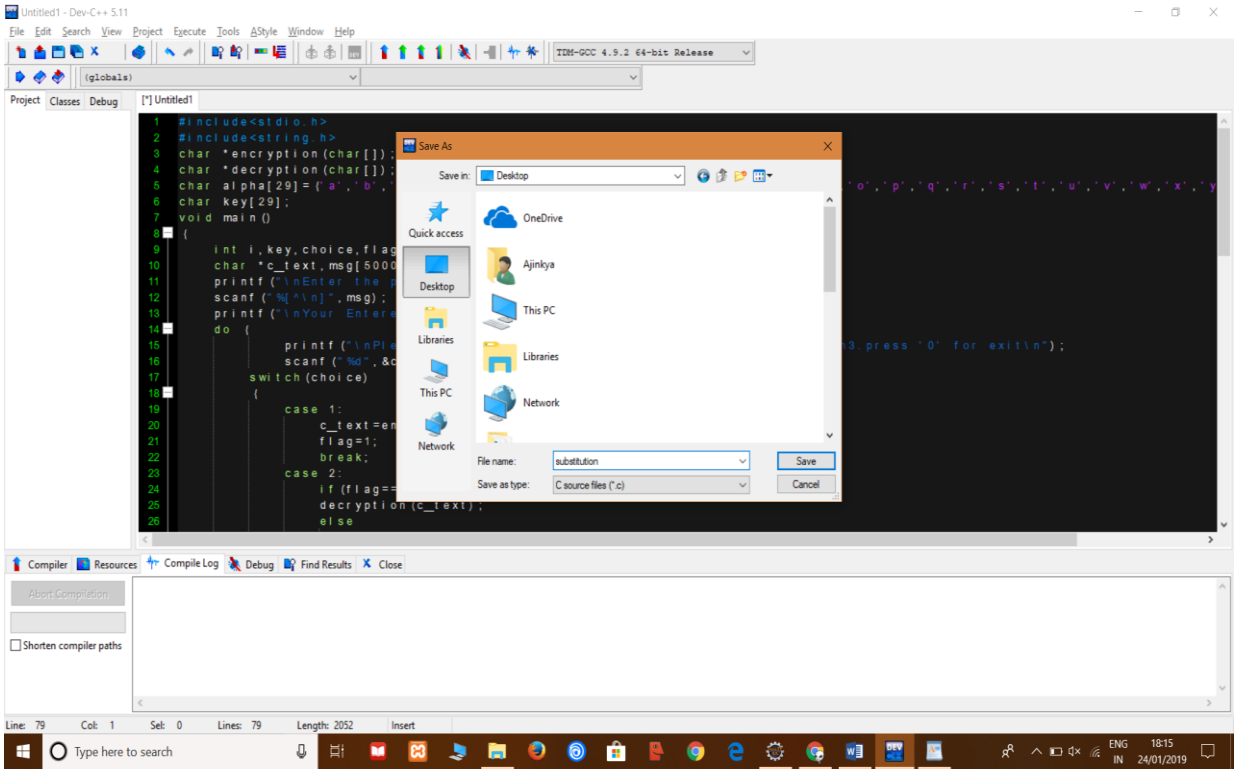


3. Copy paste the contents into the new ‘untitled’ c file

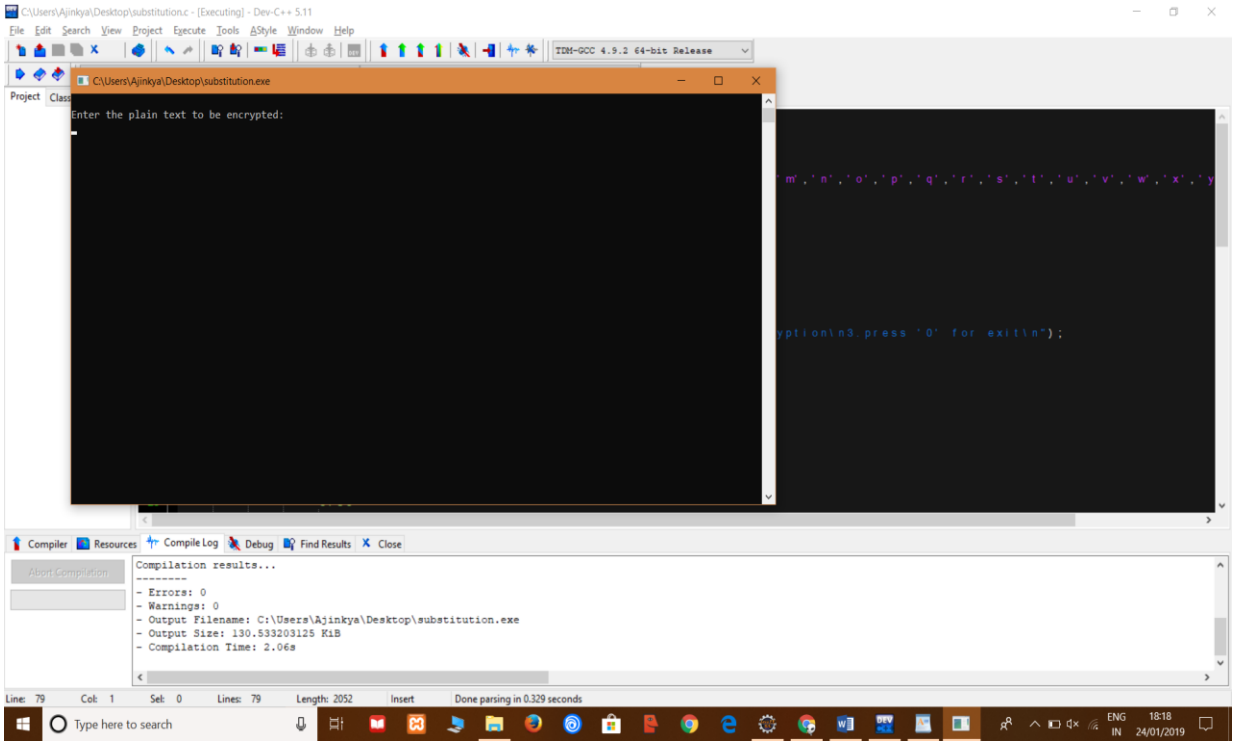


4. After pasting the content, go to Execute → Compile and run → save the file ‘filename.c’ to any location and let the code compile

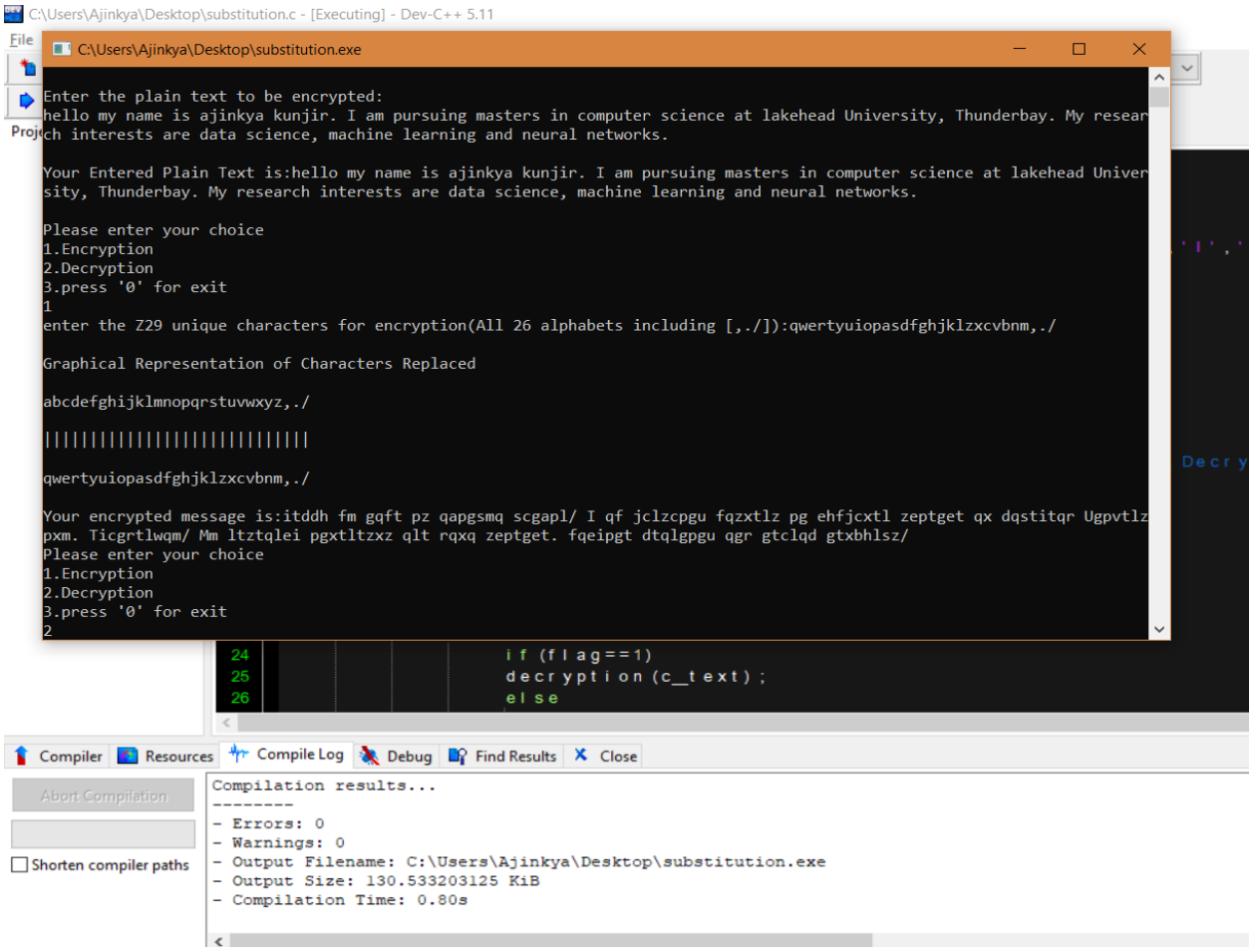




5. Watch the compilation process occurring in the bottom half ‘compile log’ and programs runs in a terminal



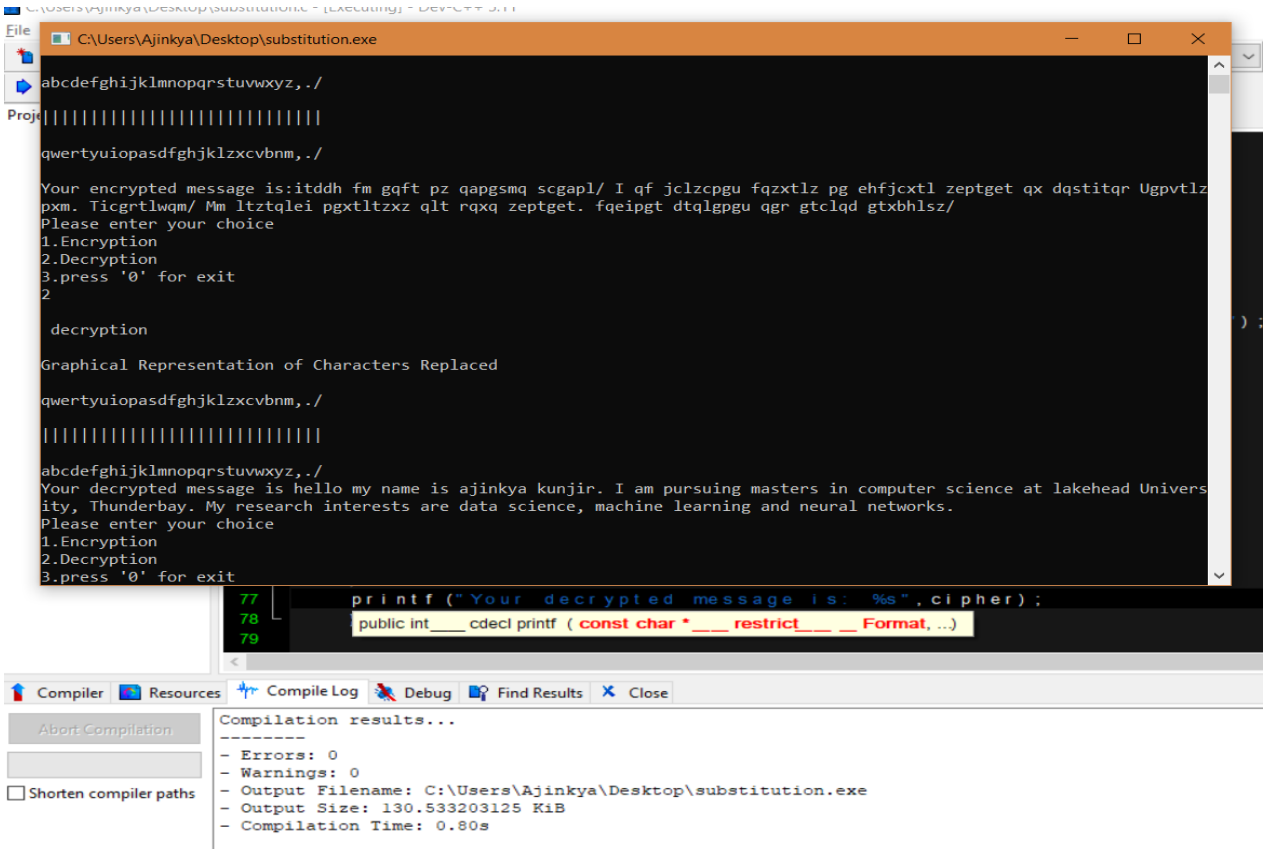
6. Output of the program



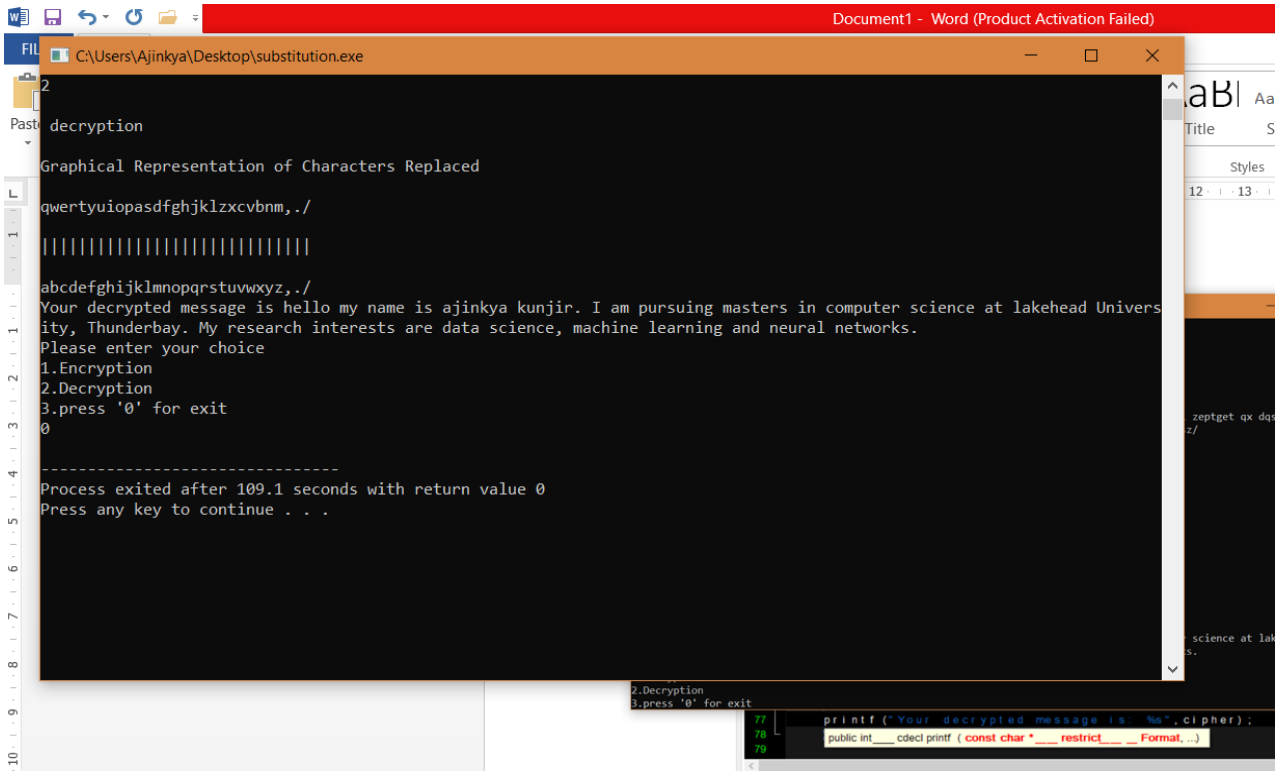
- Run→ enter the plaintext → You are given a choice of Encryption, Decryption and exit → put either ‘1’ or ‘2’ (DO NOT PUT FULL STOP AFTER I.E ‘1.’ Or ‘2.’)
- For providing Z29 values, enter a random combination of all 26 alphabets on keyboard with comma(,), fullstop(.) and hash(/). That makes 29 in total.
- Graphical representation of alphabetic mapping is shown on the terminal.
- ***Plaintext paragraph provided*:** hello my name is ajinkya kunjir. I am pursuing masters in computer science at lakehead University, Thunderbay. My research interests are data science, machine learning and neural networks.
- ***Encryption Obtained by substitution cipher***: itddh fm gqft pz qapgsmq scgapl/ I qf jclzcpgu fqzxtlz pg ehfjcxtl zeptget qx dqstitqr Ugpvtlzpxm. Ticgrtlwqm/ Mm ltztqlei pgxtltzxz qlt rqxq zeptget. fqeipgt dtqlgpgu qgr gtclqd gtxbhlsz/
- ***Key Provided (Z29):*** *qwertyuiopasdfghjklzxcvbnm,./*
- ***Decryption Obtained by Cipher***: hello my name is ajinkya kunjir. I am pursuing masters in computer science at lakehead University,

Thunderbay. My research interests are data science, machine learning and neural networks.

- Screenshot of decryption is given below:



- Finally , choose ‘3’ and exit the program



Problem 2

2. Write computer programs for the Permutation Cipher based on Z29 as in Problem 1. In encryption program, the inputs are a value of m (the size of permutation), a permutation as the key and the plaintext, and the output is the ciphertext. Write the decryption program accordingly. Try your programs by some text. Note that since m and the length of plaintext is not fixed, paddings might be added to the end of plaintext by the program. You may think about what kind padding is better for the security and design your paddings.

Solution: (folder – perm_new)

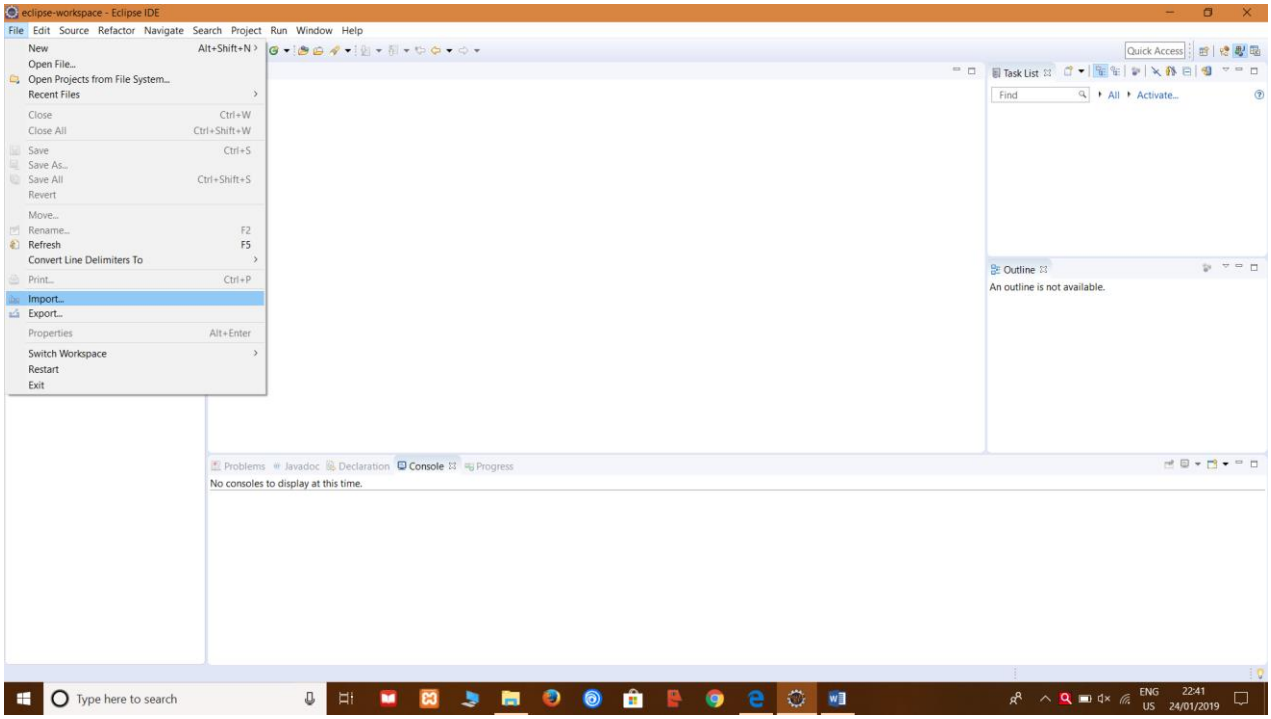
- Language of choice: Java language
- Software/IDE used: Eclipse IDE for Java
- You can download the Eclipse software from - https://www.eclipse.org/downloads/
- Download the JDK for windows from https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
- Name of the folder : perm_new

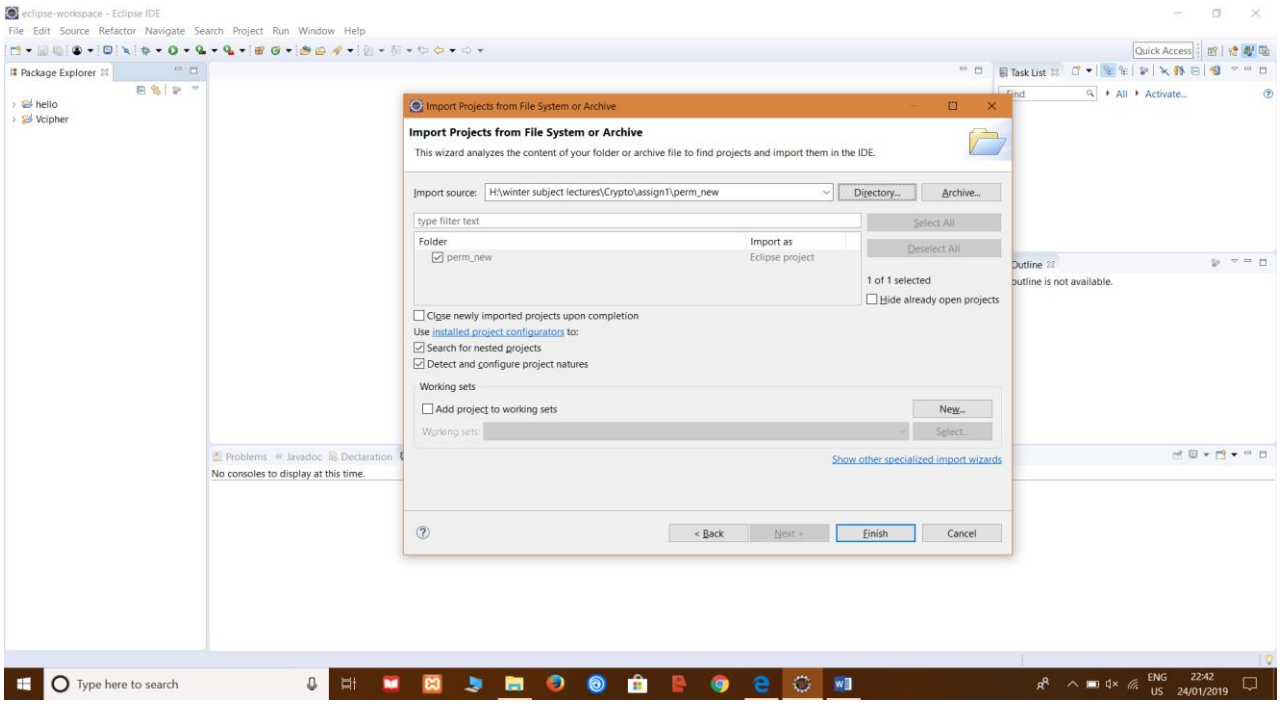
➢ *Steps to follow:*

1. Download Eclipse IDE and JDK from the websites mentioned above and set the environment of Eclipse IDE to open perspective – Java.

2. Follow the screenshots given below for importing and running the project

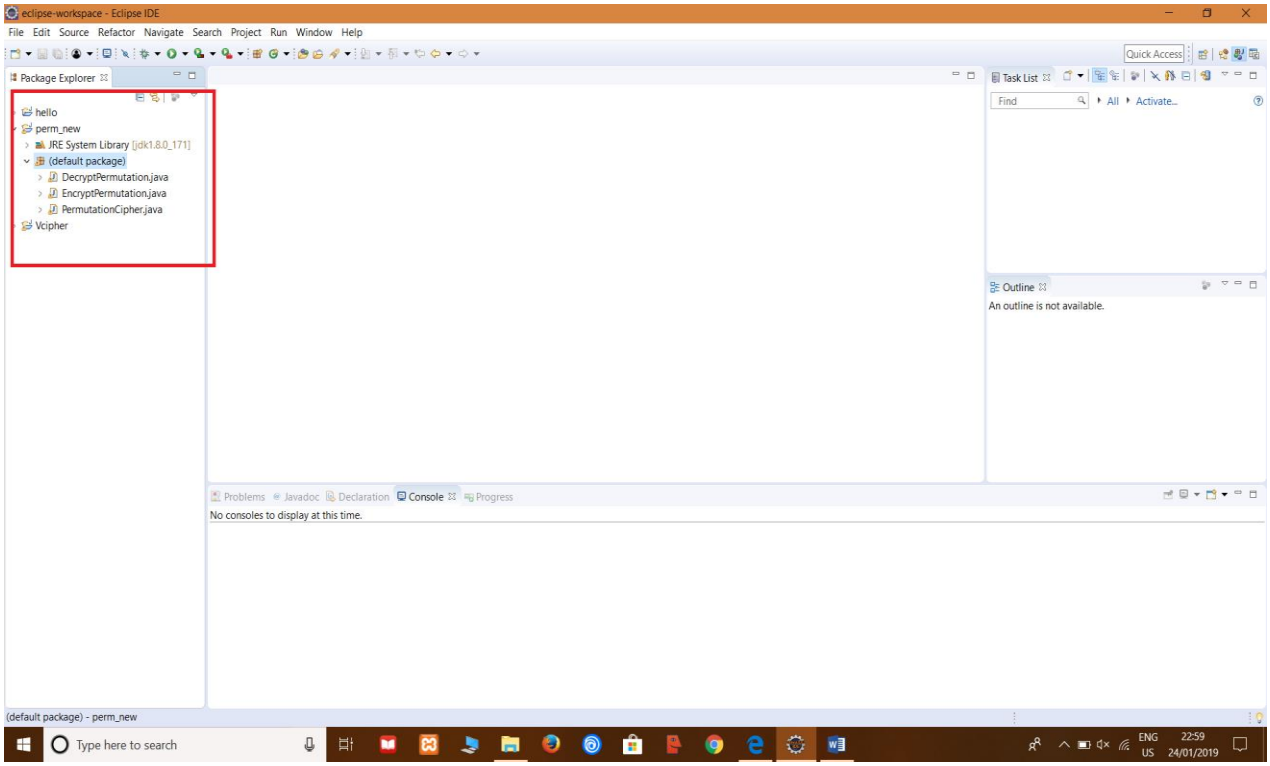
1. Open eclipse IDE → Import, click on Import and import the folder by selecting ‘projects from archive’





2. After selecting the directory, you will see the contents in the package explorer window. Refer the image given below

3. The program saves the ‘m’ sized key in the ‘output_key.txt’ file after generation, hence delete the txt. File if present while importing in the folder.



4. Now , left click on the folder in package explorer and select ‘Run as’ → Java application
5. Keep your eyes on the console window at the bottom
6. Follow the instructions and enter the plaintext

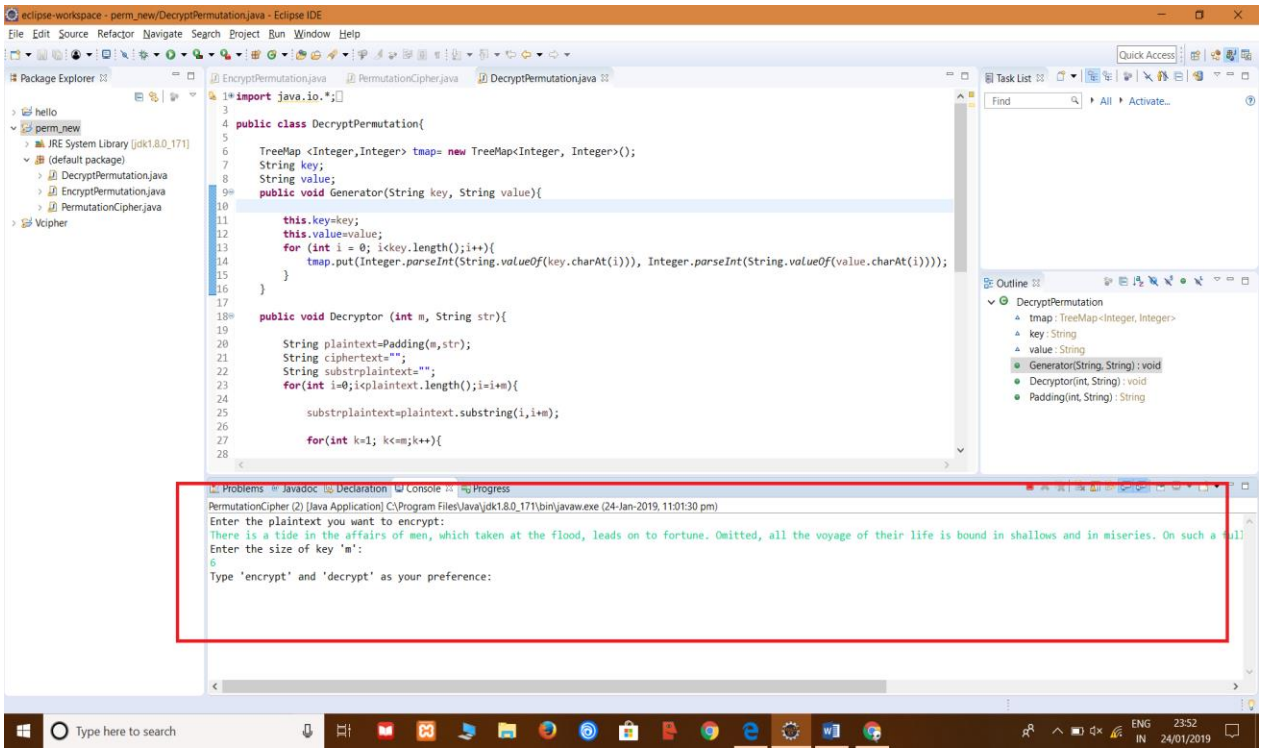
Plaintext Entered: There is a tide in the affairs of men, which taken at the flood, leads on to fortune. Omitted, all the voyage of their life is bound in shallows and in miseries. On such a full sea are we now afloat. And we must take the current when it serves, or lose our ventures.

7. Put the plaintext, enter the size of key ‘m’ (size can be any 5,6,7,8….so on)

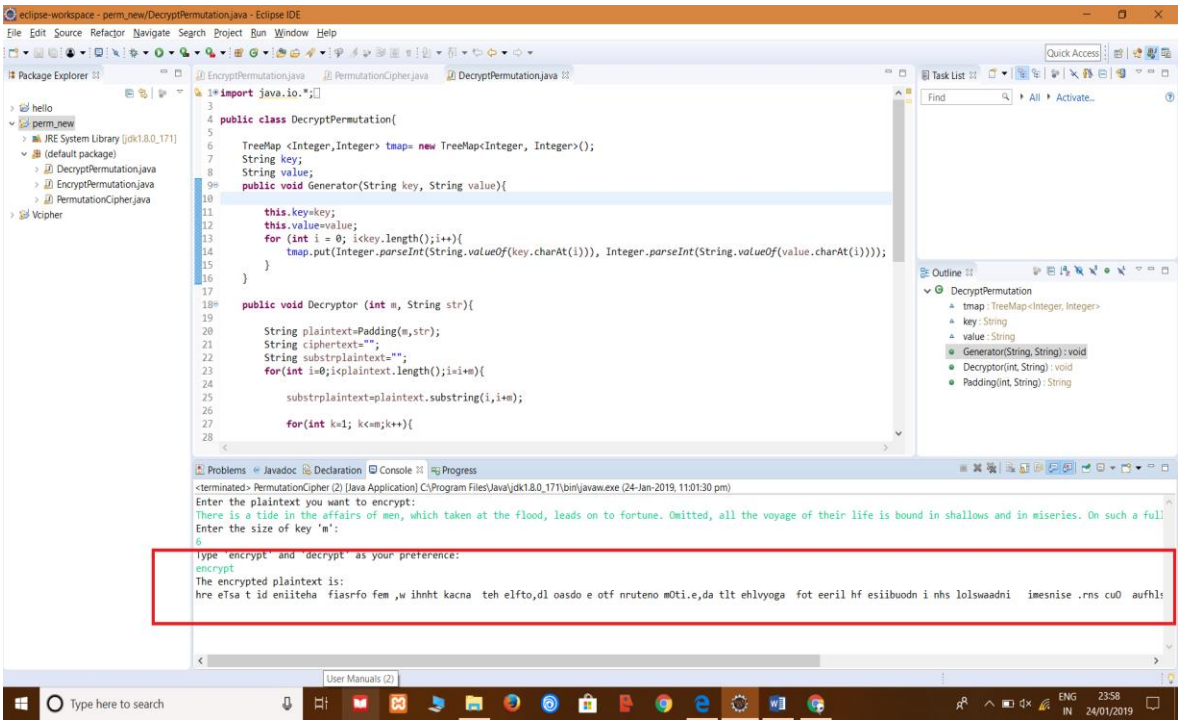
‘m’ = 5

‘m’ = 6

‘m’ = 7…….Any key number

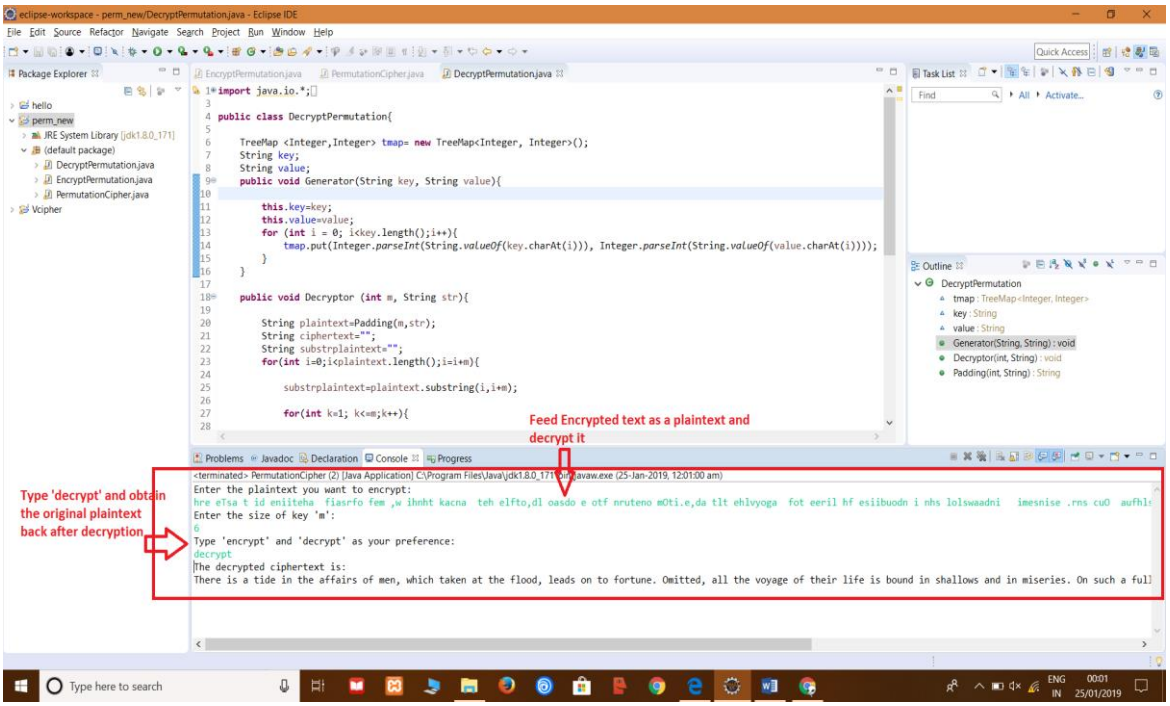


8. Type ‘encrypt’ or ‘decrypt’ based on the preference. If you type ‘decrypt’ then the program will ask you to encrypt the data first.



Encrypted Plaintext: hre eTsa t id eniiteha fiasrfo fem ,w ihnht kacna teh elfto,dl oasdo e otf nruteno mOti.e,da tlt ehlvyoga fot eeril hf esiibuodn i nhs lolswaadni imesnise .rns cuO aufhls aelaerw on wefoltaa nA d.em suw atekttehc rertnuweh n ts reie,so v olesroruv nutere.

9. You have to copy the encrypted text from the console, re-run the program and choose to 'decrypt' now. Insert the encrypted text (output of encryption) as the input for decryption.
10. *Note: Size of key 'm' doesn't matter much, you can enter any number suitable to you. (I have used 6 in this example)*



11. You obtain the original plaintext back after decrypting the ciphertext as shown in the image above.

Problem 4. (For CS 5413 students only)

Read "2.5 The Vegene`re Cipher" of the lecture notes posted in course web carefully and try to understand the materials in this subsection. Then solve the following problem: The following is a piece of ciphertext which is encrypted by Vegene`re Cipher.

Plaintext - cjnpkgrlilqwawbnuptgkerwxuzviaiiysxckwdntjawhqcutttvp
tewtrpgvcwlkkkgczafsihrimixukrwxrfmgfgkfxgukpjvvzmcmj
vawbnuptgcicvxvkgczkekgcqbchvnrqhhwiadfrcyxgvzqqtuvbd
guvttkccdpvvfphftamzxqwrtgukcelqlrxgvycwtncbjkkeerecj
qihvrjzpkkfexqgjtpjfupemswwxcjqxzpjtxkvlyaeaemwhovudk
mnfxegfrwxtdyiaecyhlgjfpogymbxyfpzxxvpngkxfitnkfdniyr
wxukssxpkqabmvkgcqbciagpadfrcyxgvyyimjvwpkgscwbpurwxq
kftkorrwvnrqhxurlslgvjxmvccraceathhtfpmeygczwgutttvtt
katmcvgiltwcsmjmvyghitfzaxodkbf

Try to find the plaintext. You need to write programs for computing index of coincidence Ic and mutual index of coincidence MIc. Use the methods discussed in the lecture notes to find the key size m first, and then find out the key. Using the found key you are able to decrypt the ciphertext.

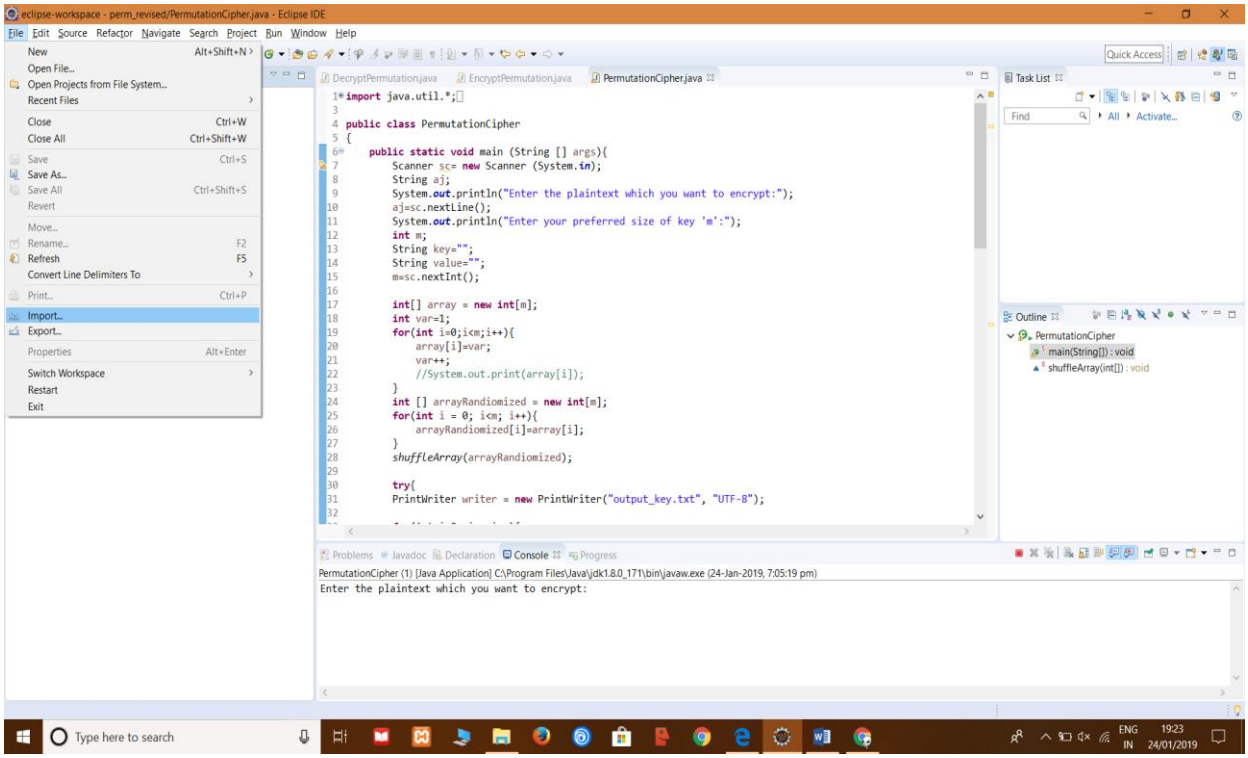
Solution: (folder – Vcipher, file- Vcipher.java)

- Language of choice: Java language
- Software/IDE used: Eclipse IDE for Java
- You can download the Eclipse software from - https://www.eclipse.org/downloads/
- Download the JDK for windows from https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
- Name of the folder-file: Vcipher/Vcipher.java

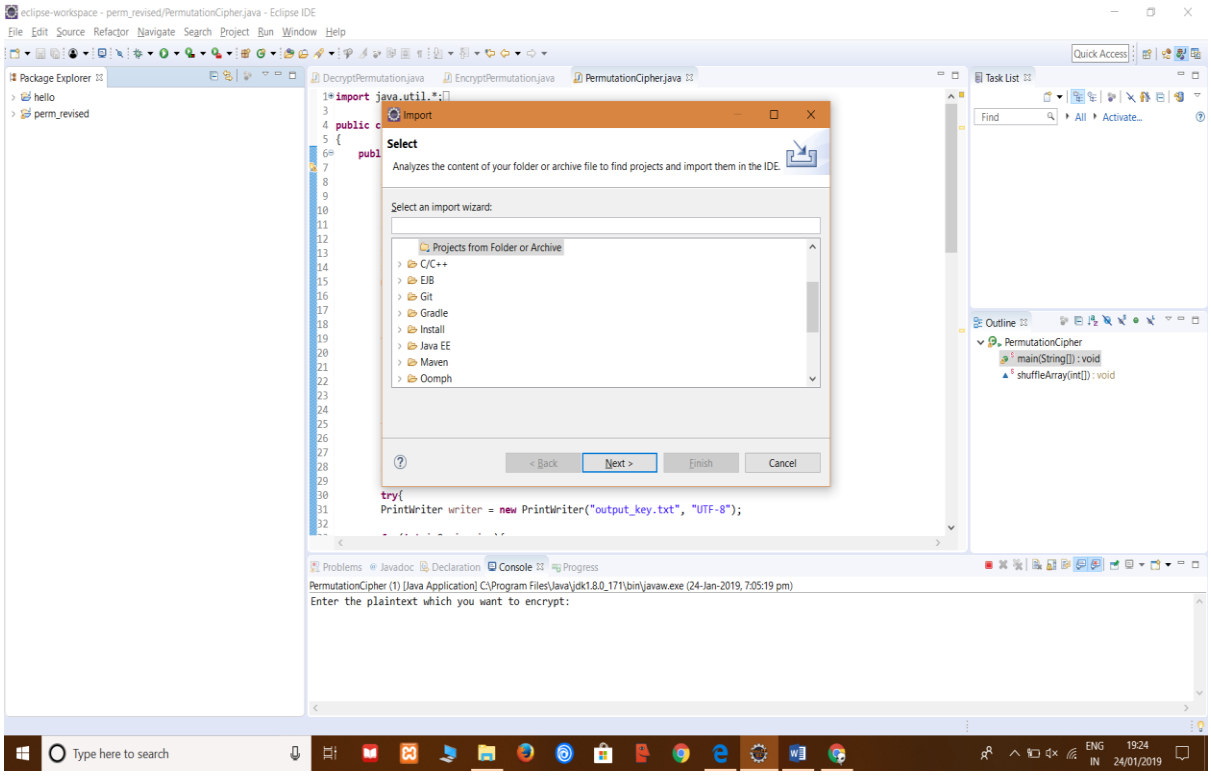
➢ *Steps to follow:*

1. Download Eclipse IDE and JDK from the websites mentioned above and set the environment of Eclipse IDE to open perspective – Java.

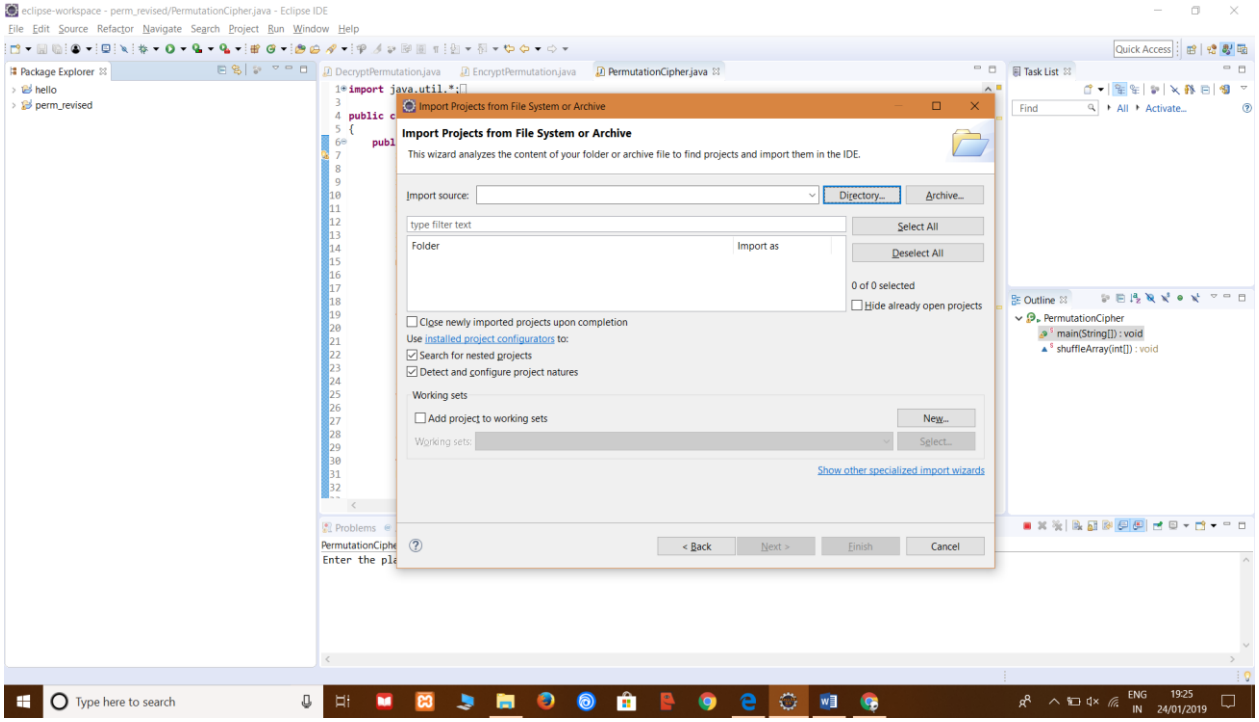
2. Follow the screenshots given below for importing and running the project

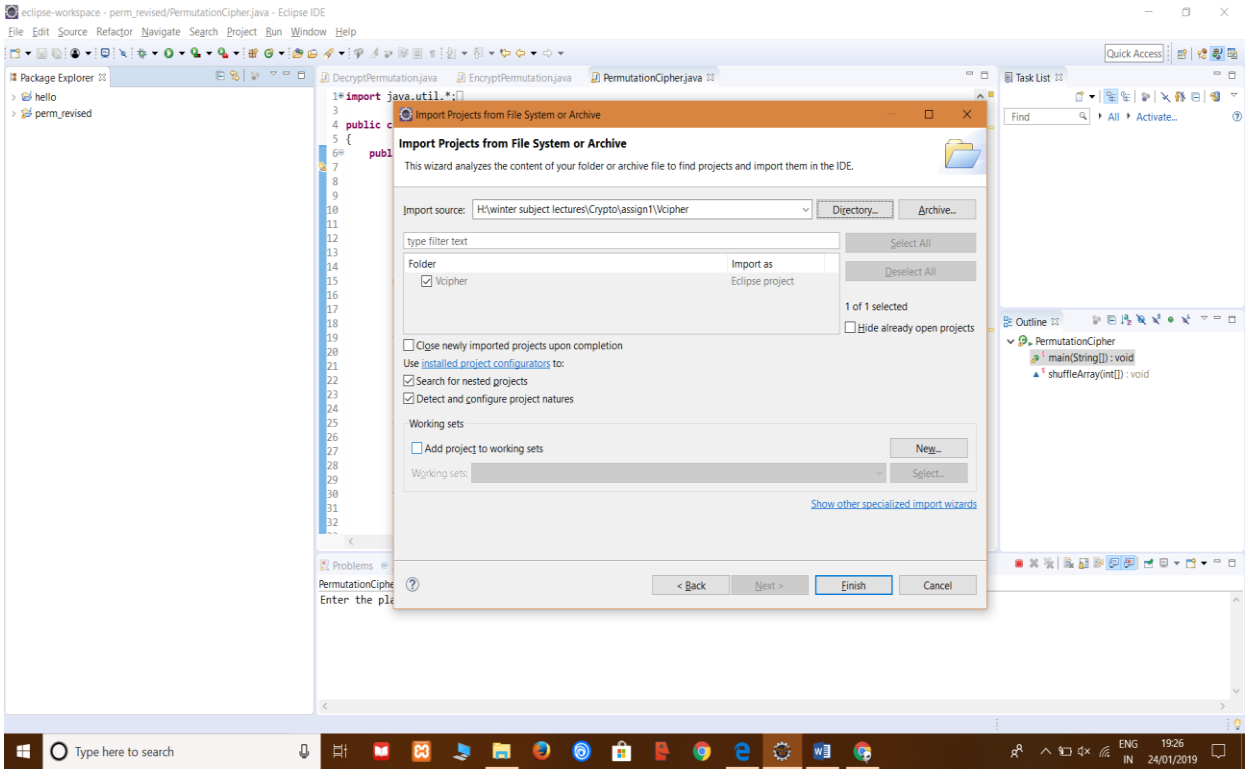


3. Go to Projects from Folder on Archive in Import wizard and proceed by selecting the folder provided with this draft i.e Vcipher

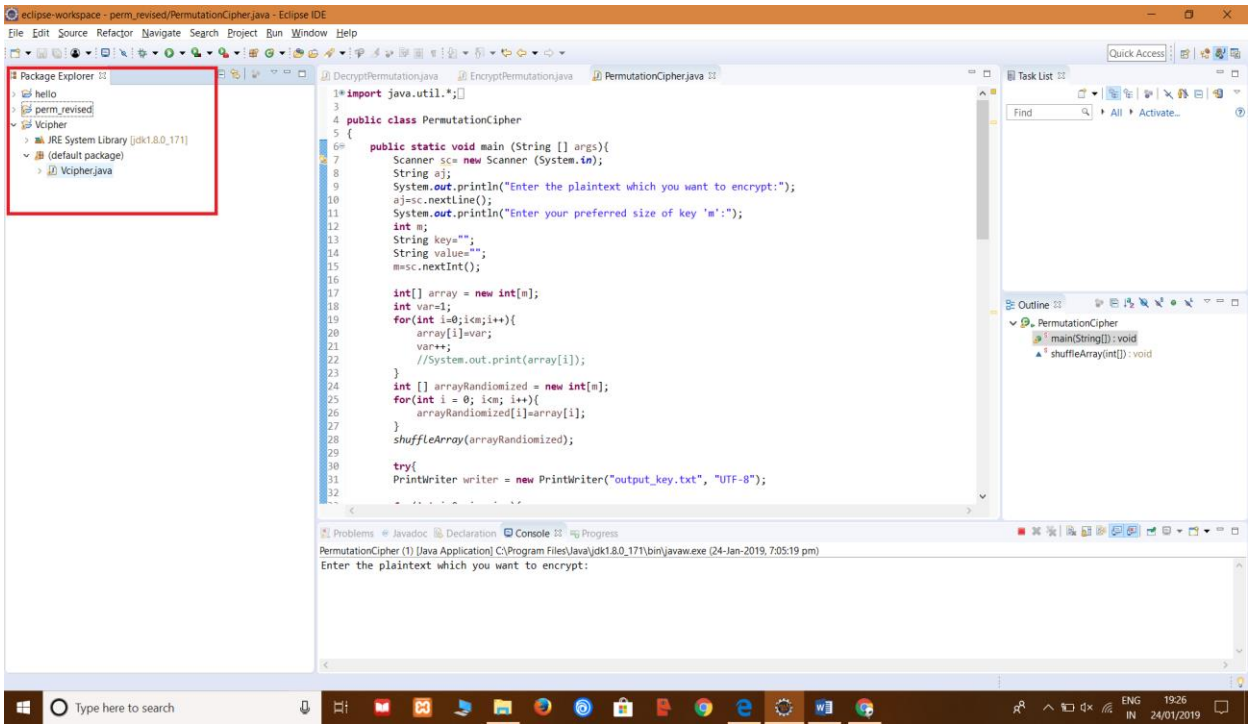


4. Select directory → location

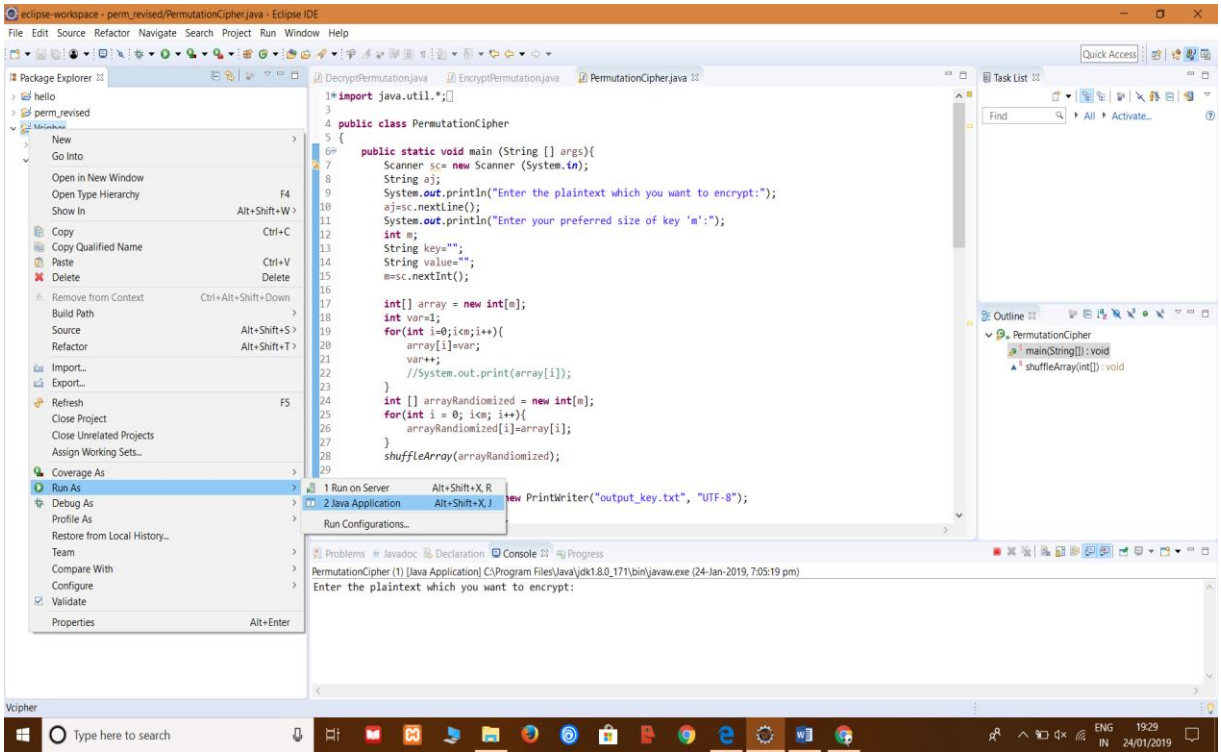




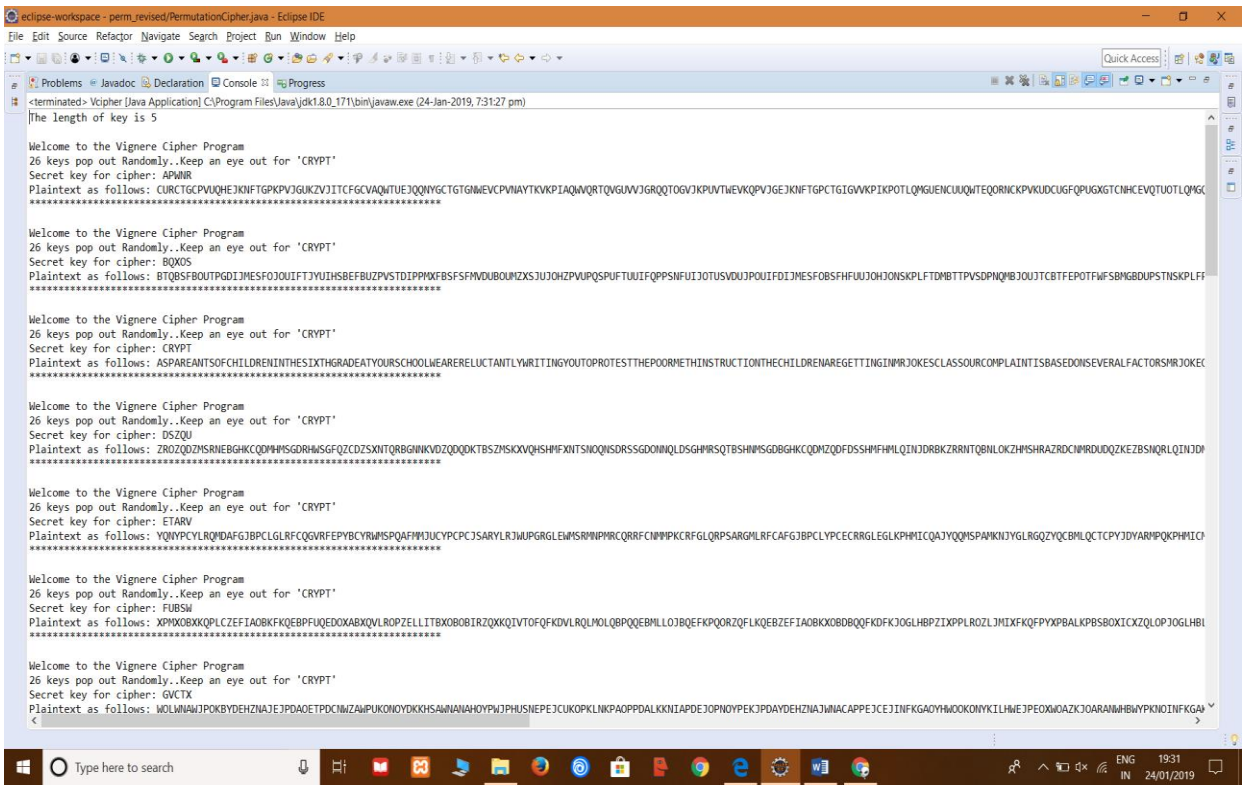
5. You will see the imported folder in the package explorer of Eclipse as shown below.



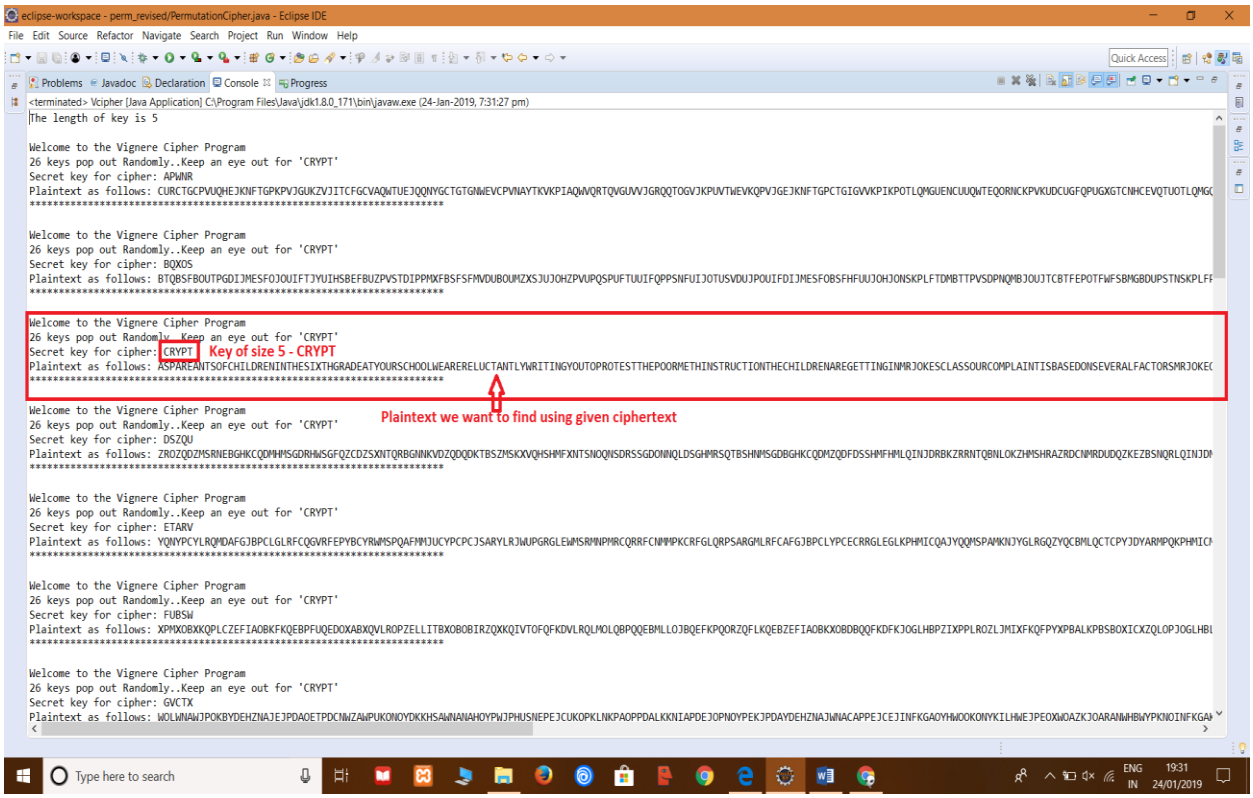
6. Right click on Vcipher→run as → java application



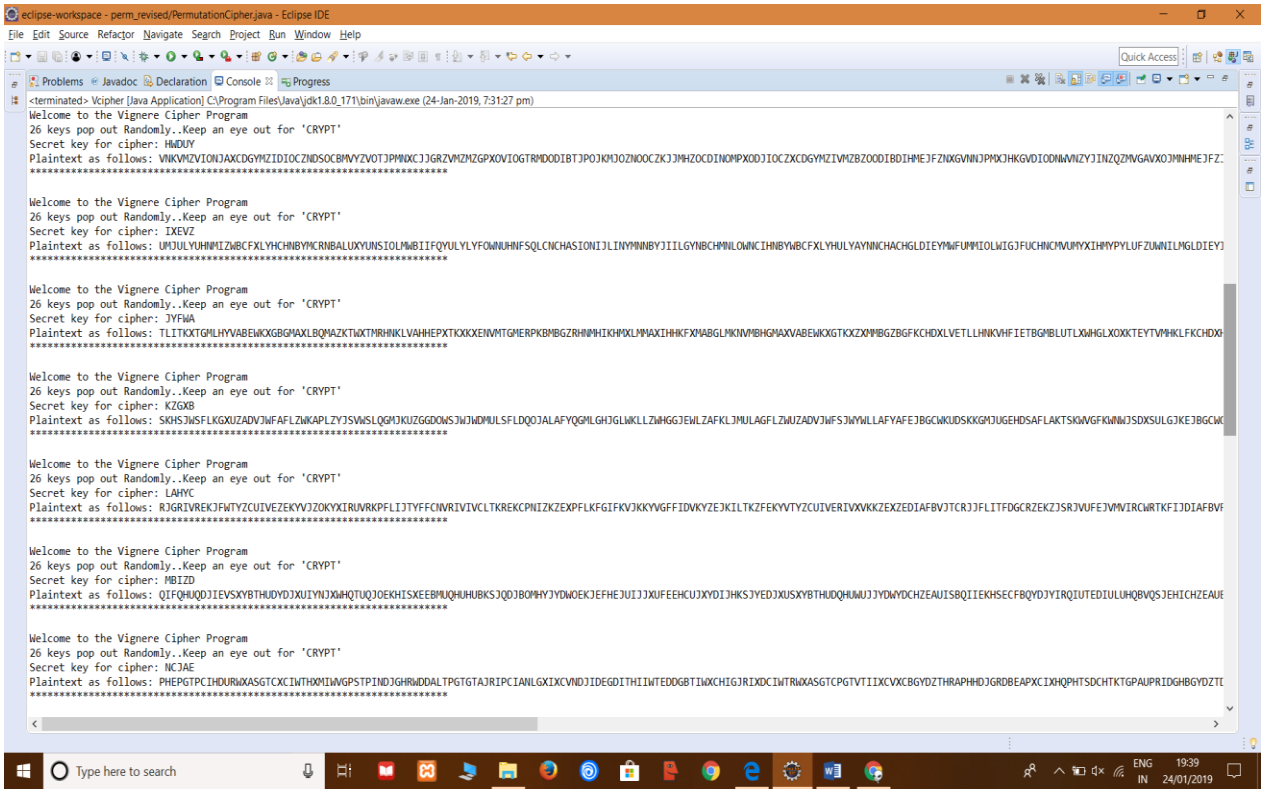
7. You can watch the output in the console at the bottom of the IDE

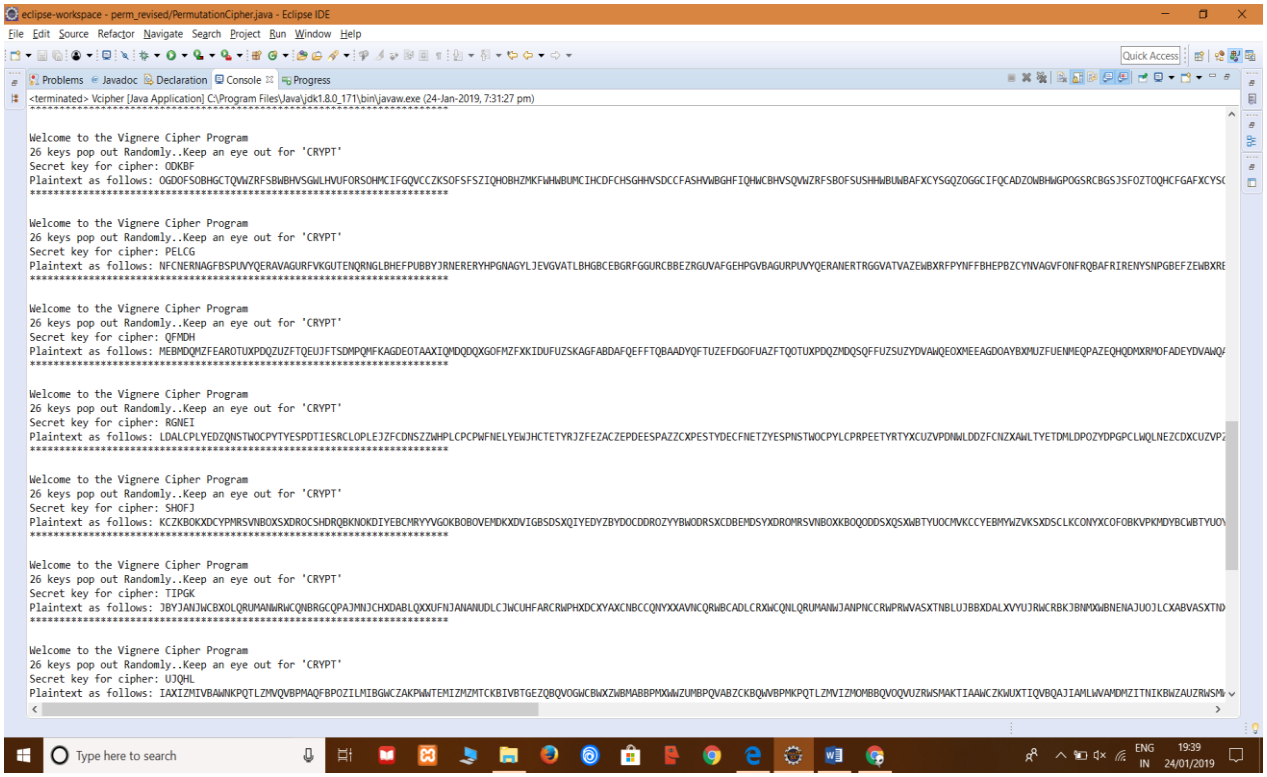


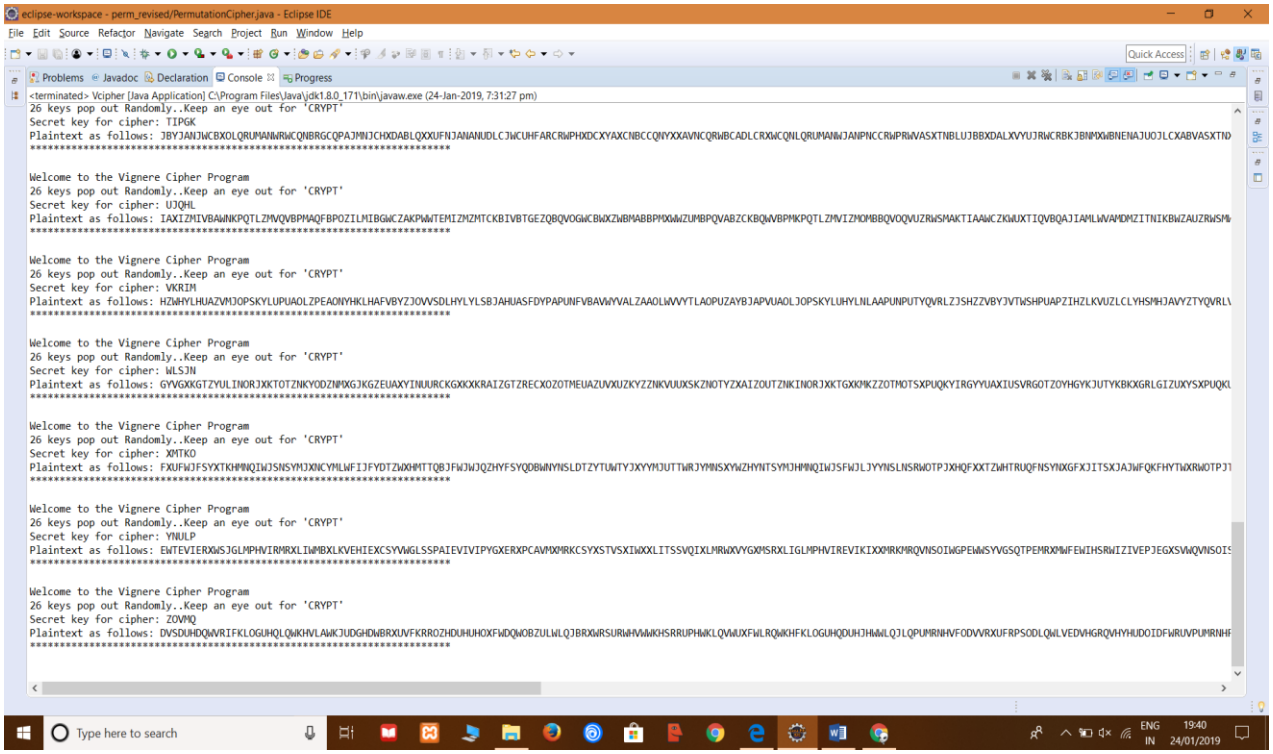
8. There are 26 keys popping out randomly in the output with all possible combinations. The length of key coded and found is 5 which is displayed at the start of the output.
9. Using this found key of size m=5, the plaintext is found from the given ciphertext in the assignment problem.



10. The key ‘CRYPT’ is seen in first few iterations of the output. The screenshot given below displays rest of the keys out of 26.


eclipse-workspace - perm_revised/PermutationCipher.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Problems Javadoc Declaration Console Progress
<terminated> Vcipher [Java Application] C:\Program Files\Java\jdk1.8.0_171\bin\javaw.exe (24-Jan-2019, 7:31:27 pm)
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: HWDUY
Plaintext as follows: VNKVMZVIONJAXCDGYMZIDIOCZNDSOCBMVYZVOTJPMNXCJJGRZVMZMZGPXOVIOGTRMDODIBTJPOJKMJOZNOOCZKJJMHZOCDINOMPXODJIOCZXCDGYMZIVMZBZOODIBDIHMEJFZNXGVNNJPMXJHKGVDIODNWVNZYJINZQZMVGAVXOJMNHMEJFZ
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: IXEVZ
Plaintext as follows: UMJULYUHNMIZWBCFXLYHCHNBYMCRNBALUXYUNSIOLMWBIIFQYULYLYFOWNUHNFSQLCNCHASIONIJLINYMNNBYJIILGYNBCHMNLOWNCIHNBYWBCFXLYHULYAYNNCHACHGLDIEYMWFUMMIOLWIGJFUCHNCMVUMYXIHMYPYLUFZUWNILMGLDIEYJ
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: JYFWA
Plaintext as follows: TLITKXTGMLHYVABEWKXGBGMAXLBQMAZKTWXTMRHNKLVAHHEPXTKXKXENVMTGMERPKBMBGZRHNMHIKHMXLMMAXIHHKFXMABGLMKNVMBHGMAXVABEWKXGTKXZXMMBGZBGFKCHDXLVETLLHNKVHFIETBGMBLUTLXWHGLXOXKTEYTVMHKLFKCHDX
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: KZGXB
Plaintext as follows: SKHSJWSFLKGXUZADVJWFAFLZWKAPLZYJSVWSLQGMJKUZGGDOWSJWJWDMULSFLDQOJALAFYQGMLGHJGLWKLLZWHGGJEWLZAFKLJMULAGFLZWUZADVJWFSJWYWLLAFYAFEJBGCWKUDSKKGMJUGEHDSAFLAKTSKWVGFKWNWJSDXSULGJKEJBGCWK
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: LAHYC
Plaintext as follows: RJGRIVREKJFWTYZCUIVEZEKYVJZOKYXIRUVRKPFLIJTYFFCNVRIVIVCLTKREKCPNIZKZEXPFLKFGIFKVJKKYVGFFIDVKYZEJKILTKZFEKYVTYZCUIVERIVXVKKZEXZEDIAFBVJTCRJJFLITFDGCRZEKZJSRJVUFEJVMVIRCWRTKFIJDIAFBVF
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: MBIZD
Plaintext as follows: QIFQHUQDJIEVSXYBTHUDYDJXUIYNJXWHQTUQJOEKHISXEEBMUQHUHUBKSJQDJBOMHYJYDWOEKJEFHEJUIJJXUFEEHCUJXYDIJHKSJYEDJXUSXYBTHUDQHUWUJJYDWYDCHZEAUISBQIIEKHSECFBQYDJYIRQIUTEDIULUHQBVQSJEHICHZEAUI
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: NCJAE
Plaintext as follows: PHEPGTPCIHDURWXASGTCXCIWTHXMIWVGPSTPINDJGHRWDDALTPGTGTAJRIPCIANLGXIXCVNDJIDEGDITHIIWTEDDGBTIWXCHIGJRIXDCIWTRWXASGTCPGTVTIIXCVXCBGYDZTHRAPHHDJGRDBEAPXCIXHQPHTSDCHTKTGPAUPRIDGHBGYDZTI
****
Type here to search
ENG IN 19:39 24/01/2019


eclipse-workspace - perm_revised/PermutationCipher.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Problems Javadoc Declaration Console Progress
<terminated> Vcipher [Java Application] C:\Program Files\Java\jdk1.8.0_171\bin\javaw.exe (24-Jan-2019, 7:31:27 pm)
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: ODKBF
Plaintext as follows: OGDOFSOBHGCTQVWZRFSBWBHVSGWLHVUFORSOHMCIFGQVCCZKSOFSFSZIQHOBHZMKFWHWBUMCIHCDFCHSGHHVSDCCFASHVWBGHFIQHWCBHVSQVWZRFSBOFSUSHHWBUWBAFXCYSGQZOGGCIFQCADZOWBHWGPOGSRCBGSJSFOZTOQHCFGAFXCYSC
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: PELCG
Plaintext as follows: NFCNERNAGFBSPUVYQERAVAGURFVKGUTENQRNGLBHEFPUBBYJRNERERYHPGNAGYLJEVGVATLBHGBCEBGRFGGURCBBEZRGUVAFGEHPGVBAGURPUVYQERANERTRGGVATVAZEWBXRFPYNFFBHEPBZCYNVAGVFONFRQBAFRIRENYSNPGBEFZEWBXRE
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: QFMDH
Plaintext as follows: MEBMDQMZFEAROTUXPDQZUZFTQEUJFTSDMPQMFKAGDEOTAAXIQMDQDQXGOFMZFXKIDUFUZSKAGFABDAFQEFFTQBAADYQFTUZEFDGOFUAZFTQOTUXPDQZMDQSQFFUZSUZYDVAWQEOXMEEAGDOAYBXMUZFUENMEQPAZEQHQDMXRMOFADEYDVAWQF
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: RGNEI
Plaintext as follows: LDALCPLYEDZQNSTWOCPYTYESPDTIESRCLOPLEJZFCDNSZZWHPLCPCPWFNELYEWJHCTETYRJZFEZACZEPDEESPAZZCXPESTYDECFNETZYESPNSTWOCPYLCPRPEETYRTYXCUZVPDNWLDDZFCNZXAWLTYETDMLDPOZYDPGPCLWQLNEZCDXCUZVP
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: SHOFJ
Plaintext as follows: KCZKBOKXDCYPMRSVNBOXSXDROCSHDRQBKNOKDIYEBCMRYYVGOKBOBOVEMDKXDVIGBSDSXQIYEDYZBYDOCDDROZYYBWODRSXCDBEMDSYXDROMRSVNBOXKBOQODDSXQSXWBTYUOCMVKCCYEBMYWZVKSXDSCLKCONYXCOFOBKVPKMDYBCWBTYUOV
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: TIPGK
Plaintext as follows: JBYJANJWCBXOLQRUMANWRWCQNBRGCQPAJMNJCHXDABLQXXUFNJANANUDLCJWCUHFARCRWPHXDCXYAXCNBCCQNYXXAVNCQRWBCADLCRXWCQNLQRUMANWJANPNCCRWPRWVASXTNBLUJBBXDALXVYUJRWCRBKJBNMXWBNENAJUOJLCXABVASXTND
****
Welcome to the Vignere Cipher Program
26 keys pop out Randomly..Keep an eye out for 'CRYPT'
Secret key for cipher: UJQHL
Plaintext as follows: IAXIZMIVBAWNKPQTLZMVQVBPMAQFBPOZILMIBGWCZAKPWWTEMIZMZMTCKBIVBTGEZQBQVOGWCBWXZWBMABBPMXWWZUMBPQVABZCKBQWVBPMKPQTLZMVIZMOMBBQVOQVUZRWSMAKTIAAWCZKWUXTIQVBQAJIAMLWVAMDMZITNIKBWZAUZRWSM
Type here to search
ENG IN 19:39 24/01/2019



Thank You!

Name – Ajinkya Kunjir

Student ID- 0876835

Subject- Cryptography & Network Security

Assignment 1