About the algorithm, I initialize an array of lower case char from a-z called alphabet. Then, I have the find\_pw function which has a double for loop to loop through 26 char in alphabet to form a string of 2 characters and then use bcrypt.checkpw to check if it is the correct password that we are looking for. Digest1,2,3 are the encrypted password of me and other 2 students in byte and will be passed to the find\_pw function. I also have count variable to keep track of the number of guesses and pw\_found to check the number of pw found so the function can return early and we don’t have to wait till the very end. If a password matched, the function will print out the password with the students name, number of guesses and the time taken to find that password

As we can see from the result, these are the password and the name of the 3 students, we also see the number of guesses and a long time to find out the password for the three. The time is roughly about 3 to 4 mins for a pw.

Analysis - 36 points, 6 per question listed above.

How long would it take to brute force the passwords of every student in class?

*From the test, average is about 3.5 mins a student. 144 stu -> 504 mins*

How much longer would it take if the passwords were 3 letters?

So we have 26 choice for the 1st letter, 26 choice for 2nd letter for now, if we have one more letter, we will have 26 more choice for 3rd letter. So the time would take 26 times longer. So the average is 3.5 mins a student, now it will be 91 mins a student.

How much harder would it be if you didn't know the encryption method was bcrypt?

That would take even longer, nearly impossible because there are a lot of ways of encryption method, and only one of them works.

How much harder would it be if the passwords had 1-8 characters using upper and lower case letters?

If increase 1 more letter, the time taken for the hacking the pw is 26 times harder, and with upper lowercase, that time would be doubled. It gets even harder if we are not sure of the number of the character inside the password because there are more cases to try. So I would say it is much much harder.

How much harder would it be if the passwords had 8-16 characters using upper, lower case, and numbers?

Using longer password upto 16 characters with upper and lower case mixing with numbers will make it nearly impossible to hack. Each character will have 26\*2 because upper lowercase, +10 number which is 62 choice. So 62^16 = 4.7672402e+28 guesses total we have to make.

Did this assignment change how you think about password security at all?

Yes. I quite take pw security for granted and now I realized I should have make my password more customized and mixed with special character to increase the security level of it.