Simplification of Passwords and Cracking:

Using the brute-force algorithm, it is impossible to crack the cipher within a realistic period. We also won't work on key and iv as a guess for a 128-bit string can kill any traditional computer. To make your program practical for testing, we restrict the password significantly. We suppose that the length of the password is **FIVE and it consists of digits and English letters (lower and upper) only**. For example, XYs12 or 4U0az are valid passwords for attempts.

We will use similar rules for the benchmark of our coursework, and I'll publish these rules later.

You can use OpenSSL to generate your own ciphertext with the above rule, or you can download the plaintext and ciphertext from BB for testing.

We provide an example program (crackaes.c) for cracking AES passwords with FIVE symbols only. You can copy and paste the plaintext and ciphertext into the right places in the program, or read in from files.

Note that you can consider a changing for the dictionary for different kinds of passwords later, or change it to crack passwords with any lengths or designate lengths. The flexibility for adapting various rules may contribute extra marks

Note that when insert plaintext or ciphertext into the program, do not forget to add '\n' by the end as shown in our example. Otherwise, it may cause errors in decryption.

To compile the code with the OpenSSL library, you need to use 'lcrypto' to link OpenSSL with your code. For example, to compile crackaes.c, you should use the following command

```
gcc crackaes.c -lcrypto
```