

B10615055 潘禎佑

建置環境： Windows 10 Professional / Visual Studio C++

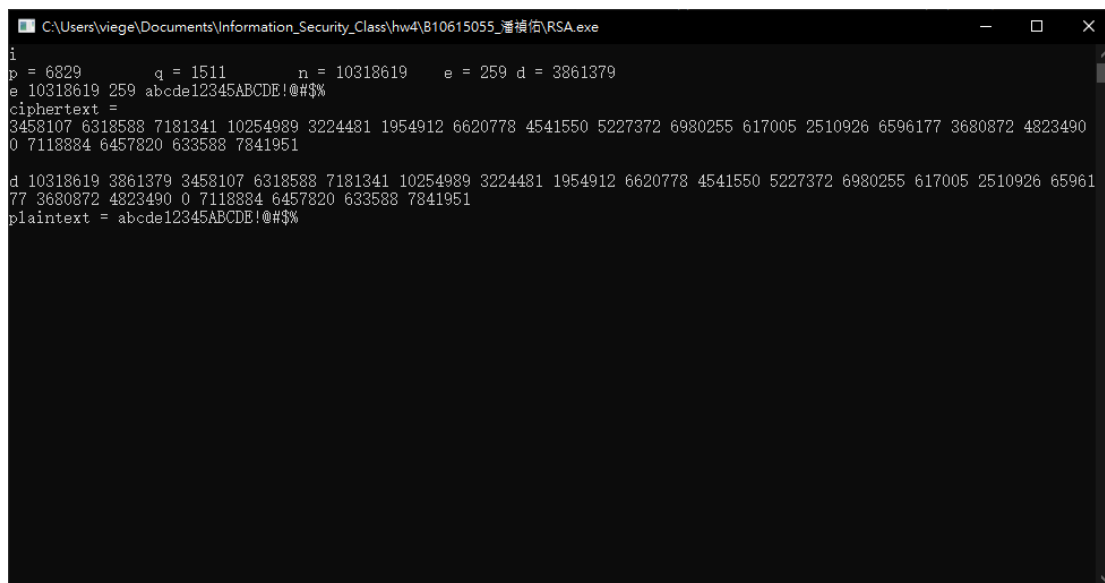
操作方式：

輸入 **i** 進行生成。

輸入 **e** 接著參數 **n e** 以及明文進行加密。

輸入 **d** 接著參數 **n d** 以及密文進行解密。

執行結果圖：



```
C:\Users\viege\Documents\Information_Security_Class\hw4\B10615055_潘禎佑\RSA.exe
i
p = 6829      q = 1511      n = 10318619      e = 259 d = 3861379
e 10318619 259 abcde12345ABCDE!@#$$%
ciphertext =
3458107 6318588 7181341 10254989 3224481 1954912 6620778 4541550 5227372 6980255 617005 2510926 6596177 3680872 4823490
0 7118884 6457820 633588 7841951
d 10318619 3861379 3458107 6318588 7181341 10254989 3224481 1954912 6620778 4541550 5227372 6980255 617005 2510926 65961
77 3680872 4823490 0 7118884 6457820 633588 7841951
plaintext = abcde12345ABCDE!@#$$%
```

程式碼解說：

在最上面有參數 `nop` 表示 `number of primes`，我先找到了前 9592 個質數存成陣列了，目前暫時使用前 1000 個，測試過大於這個數字後再生成會慢許多。

加密時，取得參數後對明文用 `square and multiply` 加密後輸出。

解密時，取得參數後，藉由分割後續字串直到換行取得密文，之後再將密文以 `square and multiply` 解密後輸出。

生成時，隨機在前 `NOP` 個質數取 `p` 與 `q`，定值 `n` 與  $pn = (p-1)*(q-1)$ ，再從 `pn` 中找 `e`，每次找到都給值，但 2% 機率退出迴圈停止找 `e`，然後找到 `d` 以後全部輸出。

遇到困難與心得：

由於大質數的處理沒有合適的想法所以這次只完成小數字的部分，在一開始直接將參數定為 `ppt` 中的範例，但在完成加密時搞錯輸出，還以為要將數字壓到可視字元中，導致加解密結果一蹋糊塗，後來釐清題意後將輸出修正，但由於並未實作 `square and multiply`，只是單純照著 `ppt` 算，將 `nop` 實作進去後問題一大堆，後來才將 `square and multiply` 實作進去後才成功，在 `nop` 的大小上糾結許久以後定為 1000。數字大的話找 `d` 會很影響效率，還沒找到比較快速的算法取 `d`。