Student name: Wenhao Lu
Student number:S3810097
Class number:COSC2536

Q1

	convert into binary numbers																
Key1:	23888	101110101010000	0	1	0	1	1	1	0	1	0	1	0	1	0	0	0	0
Key2:	44567	1010111000010111	1	0	1	0	1	1	1	0	0	0	0	1	0	1	1	1
			□combine Key1 and key2 with XOR method:															
		key4:	1	1	1	1	0	0	1	1	0	1	0	0	0	1	1	1
Key3:	58991	1110011001101111	1	1	1	0	0	1	1	0	0	1	1	0	1	1	1	1
			□combine Key3 and key4 with XOR method:															
			0	0	0	1	0	1	0	1	0	0	1	0	1	0	0	0
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			0	0	0	4096	0	1024	0	256	0	0	32	0	8	0	0	0
			□convert into decimal number:															
		Master key:	5416															

q2

The frequencies of the English language are:

Table1

E	T	A	O	I	N	S	H	R	D	L	C	U	M	W	F	G	Y	P	B	V	K	J	X	Q	Z
12.7	9.1	8.2	7.5	7.0	6.7	6.3	6.1	6.0	4.3	4.0	2.8	2.8	2.4	2.4	2.2	2.0	2.0	1.9	1.5	1.0	0.8	0.15	0.15	0.10	0.07

The frequencies of the intercept are:

Table2

A	B	C	E	F	D	G	I	H	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
23	20	18	16	16	15	15	10	9	7	7	7	6	6	5	5	4	3	3	2	0	0	0	0	0	0
11.7	10.2	9.1	8.1	8.1	7.6	7.6	5.1	4.6	3.6	3.6	3.6	3.0	3.0	2.5	2.5	2.0	1.5	1.5	1.0	0.0	0.0	0.0	0.0	0.0	0.0

We see the most common cyphertext letter is a and b. From table 1 above, we guess that these two letters represent 'e' and 't' respectively, and make substitution we get:

Ciphertext:

```
EFe OtE_He FtK Oe_D ItNGDN tHH JtM G_ EFe JGKEtDRe tDJ _ tP
SDtOHe EC tKReIEtGD LFeEFeI tDMEFGDN FtK OeeD NtGDeJ CI
DCE. _ FCL G HCDN QCI EFGK LtI EC eDJ. FCL G HC_N QCI
TetRe. FCL LG_H G FtGH EFe JtM LFeD G IeESID EC EFe OCKCP
CQ PM QtPGHM. P_ JetI G FCTe EC Kee MCS.
```

Find Frequencies

Make Substitutions

Remove spaces

Options:

But we notice that 'EFe' is frequently appearing in the passage. The most common word in English is the. 'EFe' could be 'the'. so 'b' is not 't'. And the third frequency letter is 'a', so 'b' could be 'a'.

```
the Oat_He haK Oe_D IaNGDN aHH JaM G_ the JGKtaDRe aDJ _ aP
SDaOHe tC aKReItaGD LhetheI aDMthGDN haK OeeD NaGDeJ CI
DCt. _ hCL G HCDN QCI thGK LaI tC eDJ. hCL G HC_N QCI
TeaRe. hCL LG_H G haGH the JaM LheD G IetSID tC the OCKCP
CQ PM QaPGHM. P_ JeaI G hCTe tC Kee MCS.
```

We notice that haK is following by He,so we guess it could be has. Also, by looking at the frequencies again, we see the next most common letter is "C", which is probably one of "o", "i" or "n". The only one of these options that makes sense is "to", so we guess "C" is "o".

Ciphertext:

```
the Oat_He has Oe_D IaNGDN aHH JaM G_ the JGstaDRe aDJ _ aP
SDaOHe to asReItaGD LhetheI aDMthGDN has OeeD NaGDeJ oI
Dot. _ hoL G HoDN QoI thGs LaI to eDJ. hoL G Ho_N QoI
TeaRe. hoL LG_H G haGH the JaM LheD G IetSID to the OosoP
oQ PM QaPGHM. P_ JeaI G hoTe to see MoS.
```

from ciphertext line 1, we see the word "aHH" which could be all in english, so "i" was represent by "h"

```
the Oat_le has Oe_D IaNGDN all JaM G_ the JGstaDRe aDJ _ aP
SDaOle to asReItaGD LhetheI aDMthGDN has OeeD NaGDeJ oI
Dot. _ hoL G loDN QoI thGs LaI to eDJ. hoL G lo_N QoI
TeaRe. hoL LG_l G haGl the JaM LheD G IetSID to the OosoP
oQ PM QaPGlM. P_ JeaI G hoTe to see MoS.
```

Then we notice that the letter "g" was appearing many time in the passage as a single word, so we guess it could be "i" in english

Ciphertext:

```
the Oat_le has Oe_D IaNiDN all JaM i_ the JistaDRe aDJ _ aP
SDaOle to asReItaiD LhetheI aDMthiDN has OeeD NaiDeJ oI
Dot. _ hoL i loDN QoI this LaI to eDJ. hoL i lo_N QoI
TeaRe. hoL Li_l i hail the JaM LheD i IetSID to the OosoP
oQ PM QaPilM. P_ JeaI i hoTe to see MoS.
```

by looking at the frequencies again, we see the next most common letter is "D", which is probably "n". And the word "hoL" was also apearing frequently which could be "How" in english.

Ciphertext:

```
the Oat_le has Oe_n IaNinN all JaM i_ the JistanRe anJ _ aP
SnaOle to asReItain whetheI anMthinN has Oeen NaineJ oI
not. _ how i lonN QoI this waI to enJ. how i lo_N QoI
TeaRe. how wi_l i hail the JaM when i IetSIn to the OosoP
oQ PM QaPilM. P_ JeaI i hoTe to see MoS.
```

in line 1, we notice the word"JistanRe" which is probably the word "distance" in english, so we guess that "J" and "R" represent "d" and "c" respectively.

Ciphertext:

```
the Oat_le has Oe_n IaNinN all daM i_ the distance and _ aP
SnaOle to asceItain whetheI anMthinN has Oeen Nained oI
not. _ how i lonN QoI this waI to end. how i lo_N QoI
```

```
Teace. how wi_l i hail the daM when i letSIn to the OosoP
oQ PM QaPilM. P_ deaI i hoTe to see MoS.
```

We see the word "whethel", which could be "whether", so "l" is r.
in the third line, the phrase "has Oeen" which could be "has been", so "o" is "b"

Ciphertext:

```
the bat_le has be_n raNinN all daM i_ the distance and _ aP
Snable to ascertain whether anMthinN has been Nained or
not. _ how i lonN Qor this war to end. how i lo_N Qor
Teace. how wi_l i hail the daM when i retSrn to the bosoP
oQ PM QaPilM. P_ dear i hoTe to see MoS.
```

we notice the word "raNinN" which is probably "raging", "g" was represented by "n". "

Ciphertext:

```
the bat_le has be_n raging all daM i_ the distance and _ aP
Snable to ascertain whether anMthing has been gained or
not. how i long Qor this war to end. how i lo g Qor
```

```
not. _ how i long Qor this war to end. how i lo_g Qor
Teace. how wi_l i hail the daM when i retSrn to the bosoP
oQ PM QaPilM. P_ dear i hoTe to see MoS.
```

the word "daM" which is probably "day",so "m" is y
in the line4. "retuSrn" is probably the word "return", so "s" is u

Ciphertext:

```
the bat_le has be_n raging all day i_ the distance and _ aP
unable to ascertain whether anything has been gained or
not. _ how i long Qor this war to end. how i lo_g Qor
Teace. how wi_l i hail the day when i return to the bosoP
oQ Py QaPily. P_ dear i hoTe to see you.
```

"Oor" which is the word "for", so 'O' is ' f'.
in line4 ,"teace" is probably "peace ,so "t" is "p".
then we move to the last line, "faPily" which is 'family' , so 'p' is 'm'.

Ciphertext:

```
the bat_le has be_n raging all day i_ the distance and _ am
```

```
unable to ascertain whether anything has been gained or
not. _ how i long for this war to end. how i lo_g for
peace. how wi_l i hail the day when i return to the bosom
of my family. m_ dear i hope to see you.
```

The final list of substitutions is given below:

The frequencies of the intercept are:

A	B	C	E	F	D	G	I	H	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
23	20	18	16	16	15	15	10	9	7	7	7	6	6	5	5	4	3	3	2	0	0	0	0	0	0
11.7	10.2	9.1	8.1	8.1	7.6	7.6	5.1	4.6	3.6	3.6	3.6	3.0	3.0	2.5	2.5	2.0	1.5	1.5	1.0	0.0	0.0	0.0	0.0	0.0	0.0
e	a	o	t	h	n	i	r	l	d	s	w	y	g	b	m	f	c	u	p						

Finally,we type the missing letter in the passage we got :
The battle has been raging all day in the distance and I am unable to ascertain whether anything has been gained or not. How i long for this war to end. How I long for peace. How will i hail the day when I return to the bosom of my family. My dear I hope to see you.

Q3

Q3. Designing Secure Online Property Auction System using Hash Algorithm (4 Marks)

Covid-19 has changed the way we conduct business these days. This is true for property auctions as well. The Prime minister of Australia recently announced a ban on in-person auctions and open-for-inspections. Large number of sellers and property agents are opting for online auctions. Based on an article published (URL: https://www.domain.com.au/news/saturday-auctions-how-will-they-work-now-they-are-all-online-944545/), we would like to highlight few facts about the current practice in online auctions:

- "Online auctions run like a mix between a live stream and a traditional auction, with buyers registering and placing bids while watching the video as if they were there."
- "Another method involves buyers sending off bids, similar to eBay, and the time allotted for the auction is extended by five minutes every time a bid is entered."

Obviously, there are many issues with online auction, but one of the critical issues is trust – the way online bidding process is conducted. We want to make sure the online bidding process is trustworthy, and nobody can cheat to win.

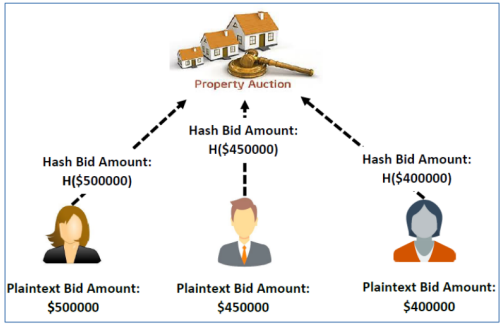

Figure-3: Cryptographic Hash Function based Online Bidding Application

Design a cheating-proof online property auction system using cryptographic hash function with the following requirements:

- A bidder can only bid with the hash value of the bid amount.
- The bidder can bid only once.
- Guessing the plaintext bid amount should be difficult.

Firstly, every bidder will receive a 6-characters-long random string from sellers such as : adYsdw
we can use this website to get a random string:https://www.random.org/strings/
Then the bidder should combine the string they received with bid amount like 500000adYsdw
then submit h(500000adYsdw) using SHA-256, it will ensure the other bidder can not guess the price they submit because they do not know the 6 random string.
there is a example :
bidder A
6-characters-long string:ofAkKO
the bid amount:400000
combination of two value, we get: 400000ofAkKO

400000ofAkKO	c3f52fae80a8fa1d87841bf77646bbc741e923326750d83923917b8be26768e8

Q4

Q4. Breaking RSA Key Faster with Multiple Servers (5 Marks)

[Note: Only for this question, you can submit the solution individually or in a group. In the case of a group submission, the maximum group members can be 3 (three), and you must mention the names of group members in the solution of this question.]

It has been found that a quantum computer with 4099 perfectly stable qubits could break the RSA-2048 encryption in 10 seconds, while a classic computer of present days requires 300 trillion years. It means, the powerful computers make the RSA cryptosystem vulnerable.

RSA cryptosystem is mainly built on the concept of prime numbers. The public-key component (***n***) of RSA cryptosystem is an integer that is the product of two prime numbers. Hence, prime factorization is a technique that can be used for breaking RSA private-key (***d***).

Prime factorization or integer factorization of a number is breaking a number down into the set of prime numbers which multiply together to result in the original number. This is also known as prime decomposition. Assume a number '77' has two prime factors. That is, '77' is a product of two prime numbers: 7 and 11 (i.e., 77 = 7 X 11).

The First 10,000 Primes
(the 10,000th is 104,729)
For more information on primes see http://primes.utm.edu/

2	3	5	7	11	13	17	19	23	29
31	37	41	43	47	53	59	61	67	71
73	79	83	89	97	101	103	107	109	113
127	131	137	139	149	151	157	163	167	173
179	181	191	193	197	199	211	223	227	229
233	239	241	251	257	263	269	271	277	281
283	293	307	311	313	317	331	337	347	349
353	359	367	373	379	383	389	397	401	409
419	421	431	433	439	443	449	457	461	463
467	479	487	491	499	503	509	521	523	541
547	557	563	569	571	577	587	593	599	601
607	613	617	619	631	641	643	647	653	659
661	673	677	683	691	701	709	719	727	733
739	743	751	757	761	769	773	787	797	809
811	821	823	827	829	839	853	857	859	863
877	881	883	887	907	911	919	929	937	941
947	953	967	971	977	983	991	997	1009	1013
1019	1021	1031	1033	1039	1049	1051	1061	1063	1069
.....									
.....									
.....									
103087	103091	103093	103099	103123	103141	103171	103177	103183	103217
103231	103237	103289	103291	103307	103319	103333	103349	103357	103387
103391	103393	103399	103409	103421	103423	103451	103457	103471	103483
103511	103529	103549	103553	103561	103567	103573	103577	103583	103591
103613	103619	103643	103651	103657	103669	103681	103687	103699	103703
103723	103769	103787	103801	103811	103813	103837	103841	103843	103867
103889	103903	103913	103919	103951	103963	103967	103969	103979	103981
103991	103993	103997	104003	104009	104021	104033	104047	104053	104059
104087	104089	104107	104113	104119	104123	104147	104149	104161	104173
104179	104183	104207	104231	104233	104239	104243	104281	104287	104297
104309	104311	104323	104327	104347	104369	104381	104383	104393	104399
104417	104459	104471	104473	104479	104491	104513	104527	104537	104543
104549	104551	104561	104579	104593	104597	104623	104639	104651	104659
104677	104681	104683	104693	104701	104707	104711	104717	104723	104729

However, a simple method to find the prime factors is to take a list of prime numbers, and start dividing a number by each prime number starting from '2' in the prime number's list. For example, first 10 prime numbers are: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29. Now, to find out the prime factors of '77', you should divide '77' by each prime number in the above list as follows unless you get another prime number as a quotient:

77 / 2 = Quotient is NOT a prime number
77 / 3 = Quotient is NOT a prime number
77 / 5 = Quotient is NOT a prime number
77 / 7 = 11 (Quotient is a prime number)

Hence, 7 and 11 are two prime factors of 77.

From the question, we know n =10772542097 and e = 95177

We can factorise n(10772542097) using wolframalpha to find p and q (link:https://www.wolframalpha.com/)

10772542097=103619*103963 these two numbers are also on the given table

then we knew the p and q, we should find out ϕ(n) because ϕ(n) = (p - 1) * (q - 1)

ϕ(n) = (103619-1)*(103963-1)=10772334516

then we should find out d because d = e^(-1) mod ϕ(n),

d=95177^(-1) mod 10772334516=3758212253

finally, we can decrypt C to get the value of M, M = C^d mod n

M=c^3758212253 mod 10772542097

we assume the ciphertext is 3389680045

3389680045^3758212253 mod 10772542097=100

then we found 100 in ascii code is letter"d"

As you know from Lecture-3 and Tutorial-3, the public-key component (n) of the RSA cryptosystems is an integer that has two prime numbers. Assume that you have found the RSA public-key as: **n = 10772542097 and e = 95177**. You want to find the **private-key (d)** for the above RSA public-key.

Say, you have the list of first 10000 prime numbers as partially shown in **Figure-4**. A complete list of first 10000 prime numbers can be found in the URL: https://primes.utm.edu/lists/small/10000.txt. Assume that you have **10 computers**. How can you take advantage of the 10 computers and perform the integer factorization tasks mentioned above to break RSA faster? Explain your algorithm and show detail steps. Please note that we are not interested in any established approach found in textbooks to find prime factors. A simple brute-force method should do the work.

[https://www.quintessencelabs.com/blog/breaking-rsa-encryption-update-state-art/]