Take Home

MATH FOR COMPUTING

SAIF HADDAD

21110214

Task 1:

The importance of prime numbers in RSA:

is because of the fact that two numbers multiplied together produces a number that can only be divided into those primes (itself and 1). (Smith, 2018)

A) Generate a public/private key pair:

P=13

A=17 (The last two digits of my university ID are 14 so the nearest prime number to the 14 I choose is 17)

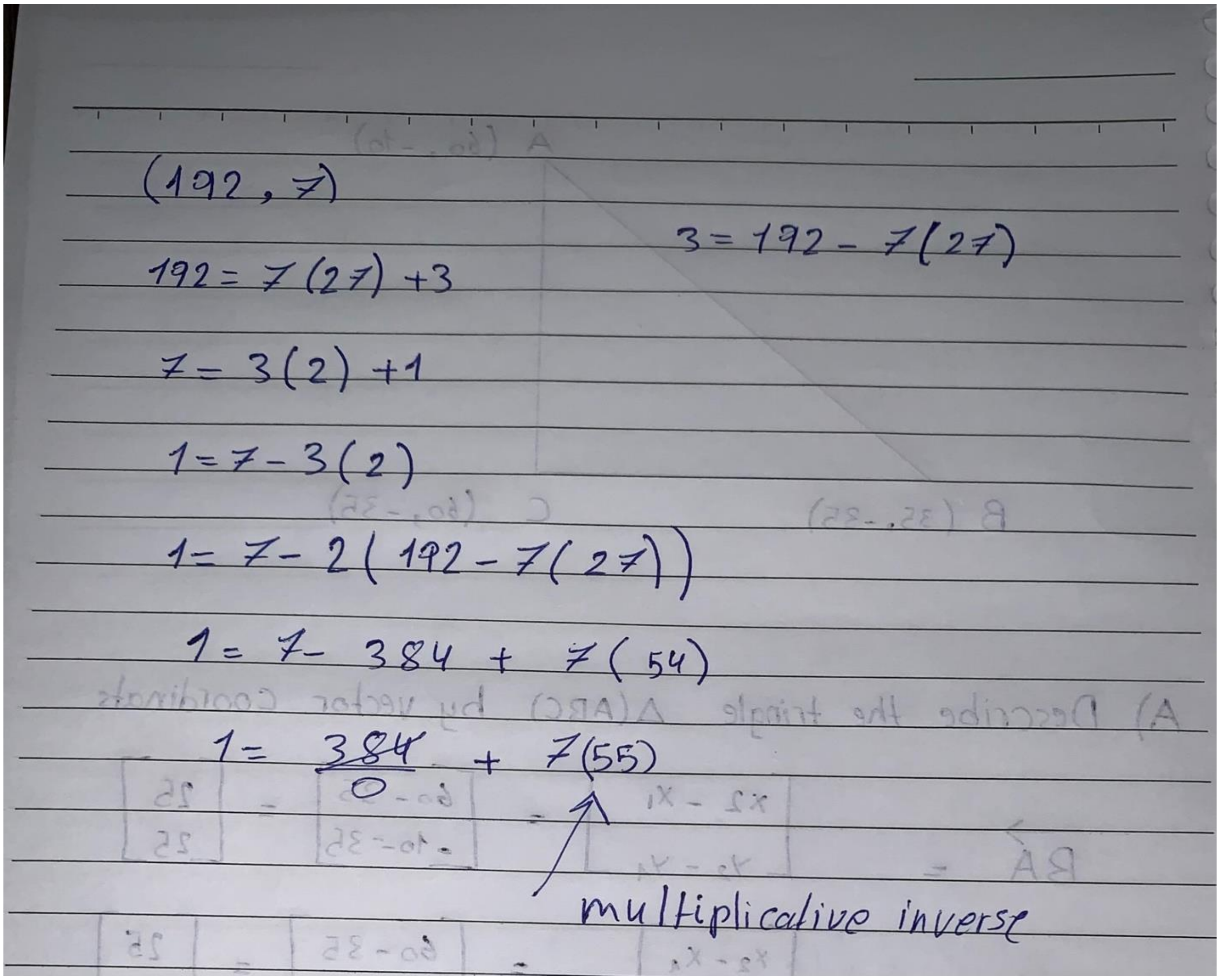
E=7 (I chose it based on e > 5)

N= P * A
N=13 * 17 = 221

qn= (P-1) * (A-1)
qn= (13-1) * (17-1)
qn= 12*16 = 192

D * E mod qn
D * 7 mod 192=1

D=55



$(192, 7)$

$3 = 192 - 7(27)$

$192 = 7(27) + 3$

$7 = 3(2) + 1$

$1 = 7 - 3(2)$

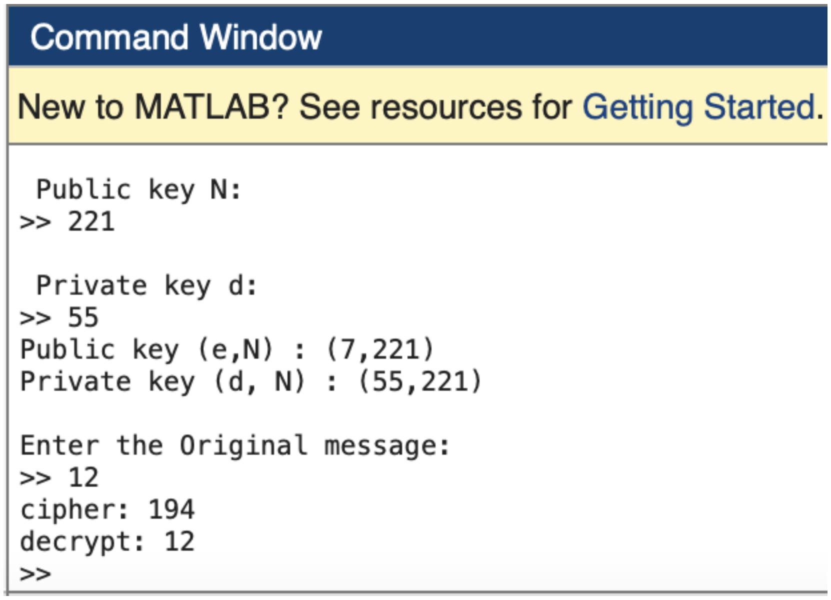
$1 = 7 - 2(192 - 7(27))$

$1 = 7 - 384 + 7(54)$

$1 = \underset{0}{\underline{384}} + 7(55)$

multiplicative inverse

public key = (E, N) = (7, 221)
private key = (D, N) = (55, 221)



```
 Public key N:
>> 221

 Private key d:
>> 55
Public key (e,N) : (7,221)
Private key (d, N) : (55,221)

Enter the Original message:
>> 12
cipher: 194
decrypt: 12
>>
```

Without using prime number:

P=18

A=20

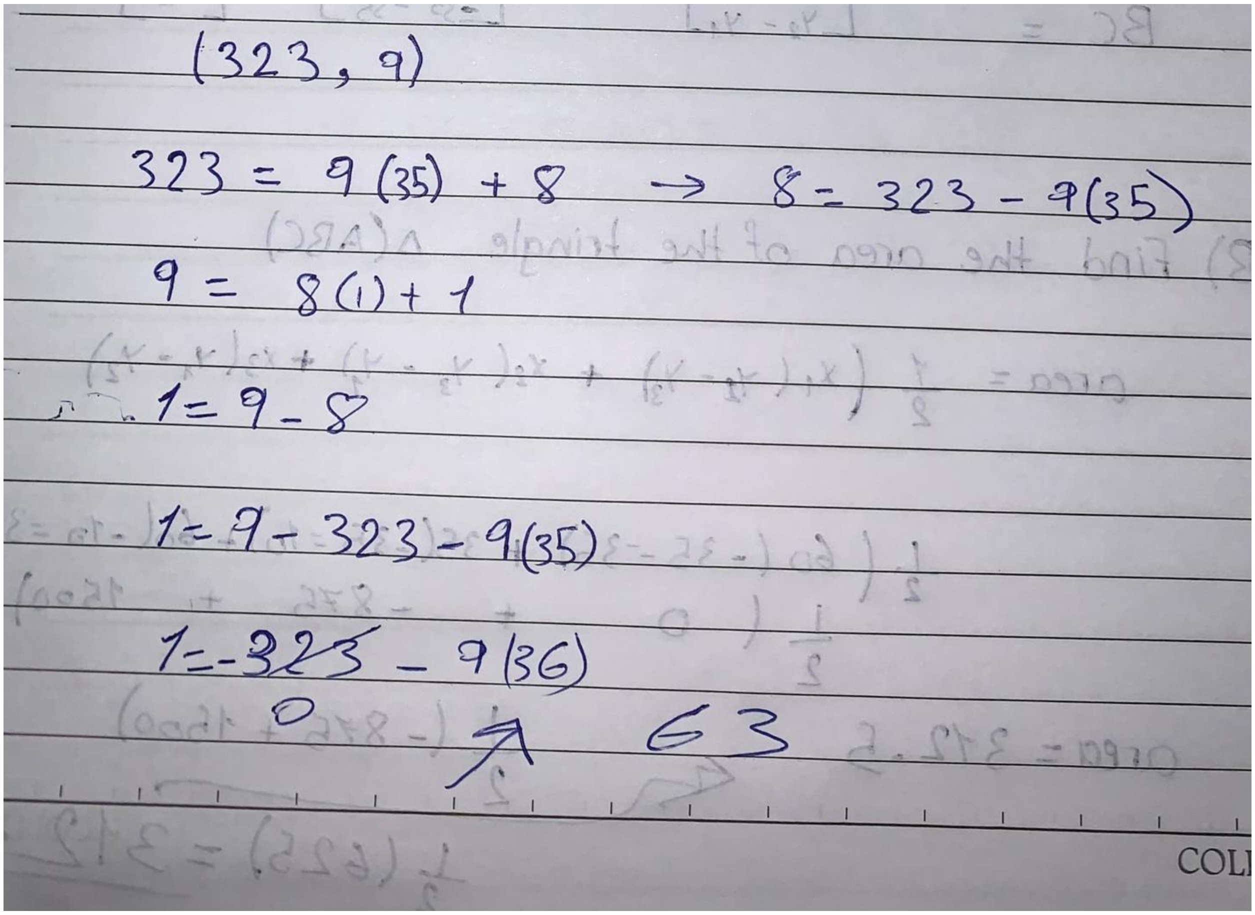
E=9

N= P * A
N= 18 * 20 = 360

qn= (P-1) * (A-1)
qn= (18-1) * (20-1)
qn= 17 * 19 = 323

D * E mod qn
D * 9 mod 323=1

D=36



$(323, 9)$

$323 = 9(35) + 8 \rightarrow 8 = 323 - 9(35)$

$9 = 8(1) + 1$

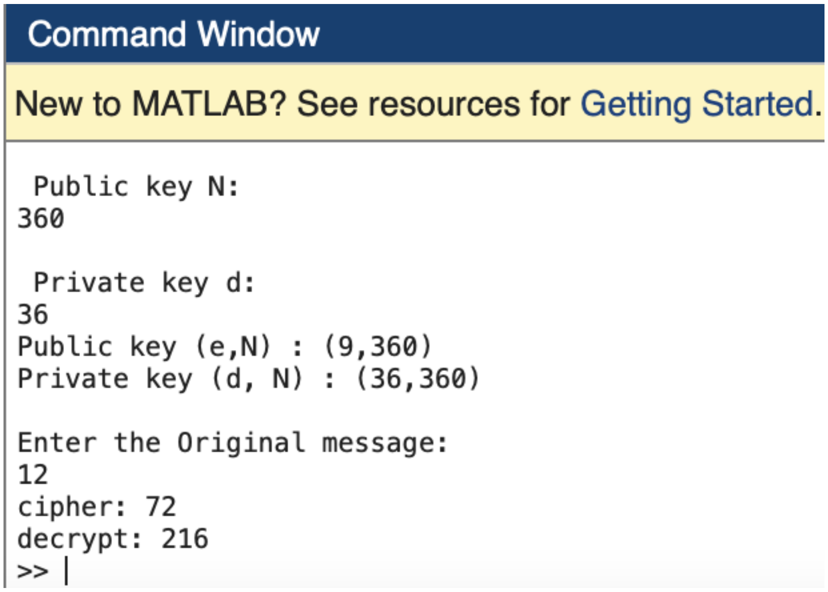
$1 = 9 - 8$

$1 = 9 - 323 - 9(35)$

$1 = -\cancel{323} - 9(36)$

0 ↗ 63

public key = (E, N) = (9, 360)
private key = (D, N) = (36, 360)

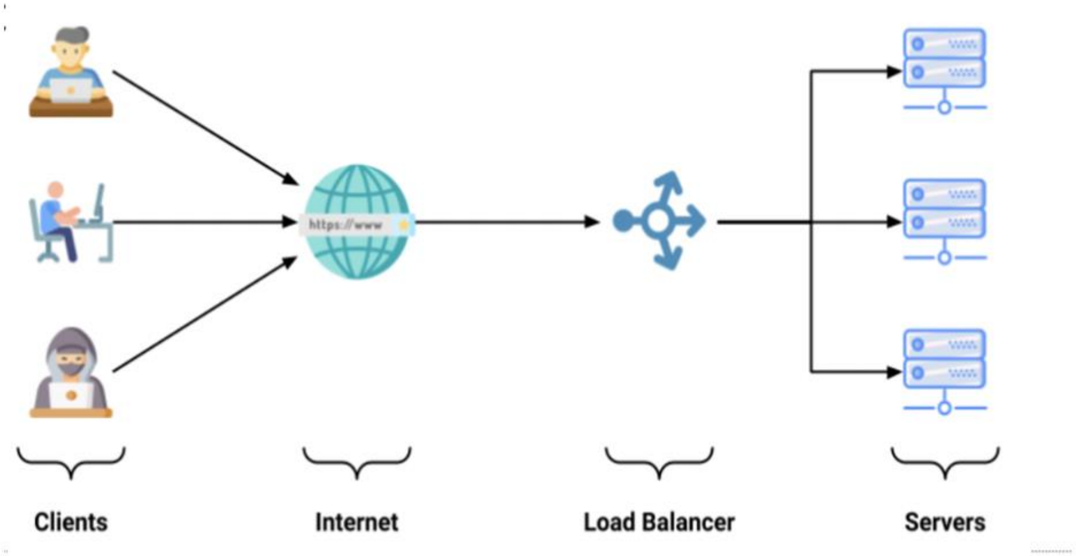


```
 Public key N:
360

 Private key d:
36
Public key (e,N) : (9,360)
Private key (d, N) : (36,360)

Enter the Original message:
12
cipher: 72
decrypt: 216
>> 
```

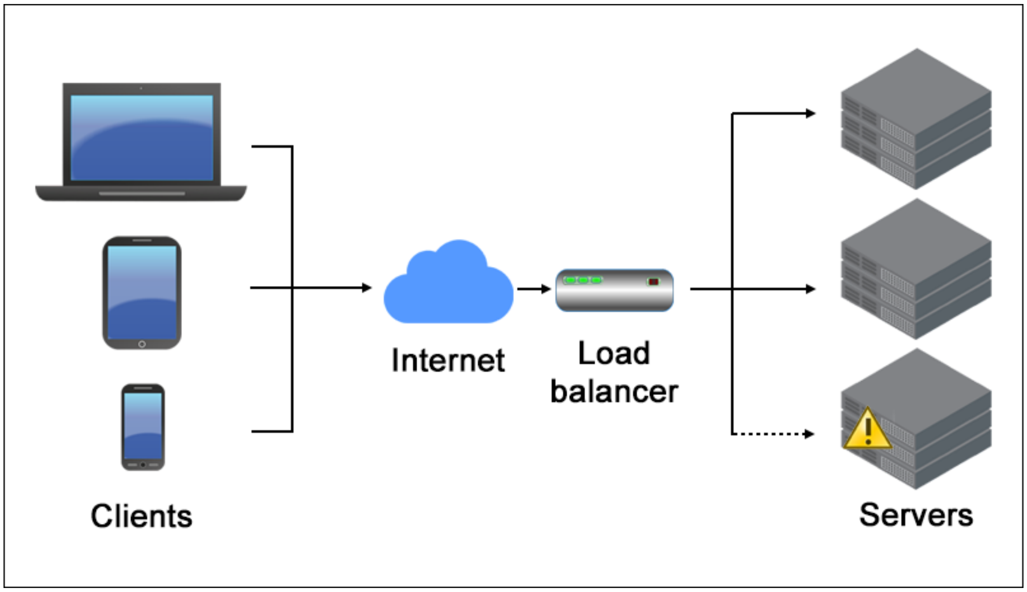
As we can see that without using primes numbers the message will change.

Task 2:

A)



From the slides



(Marijan, 2022)

Load balancing divides heavy network traffic among several servers, enabling businesses to extend horizontally to handle heavy workloads. In order to distribute the load fairly and enhance application responsiveness, load balancing distributes client requests to available servers, so boosting website availability.

c) Mid-square method

is a method for producing seemingly random numbers. In the 1940s, John von Neumann made the initial suggestion. The middle-square method's fundamental premise is to square an arbitrary seed value, then use the middle two digits of the result as the subsequent pseudo-random number. The new pseudo-random number is then used as the seed to repeat this procedure.
(GeeksforGeeks, 2018)

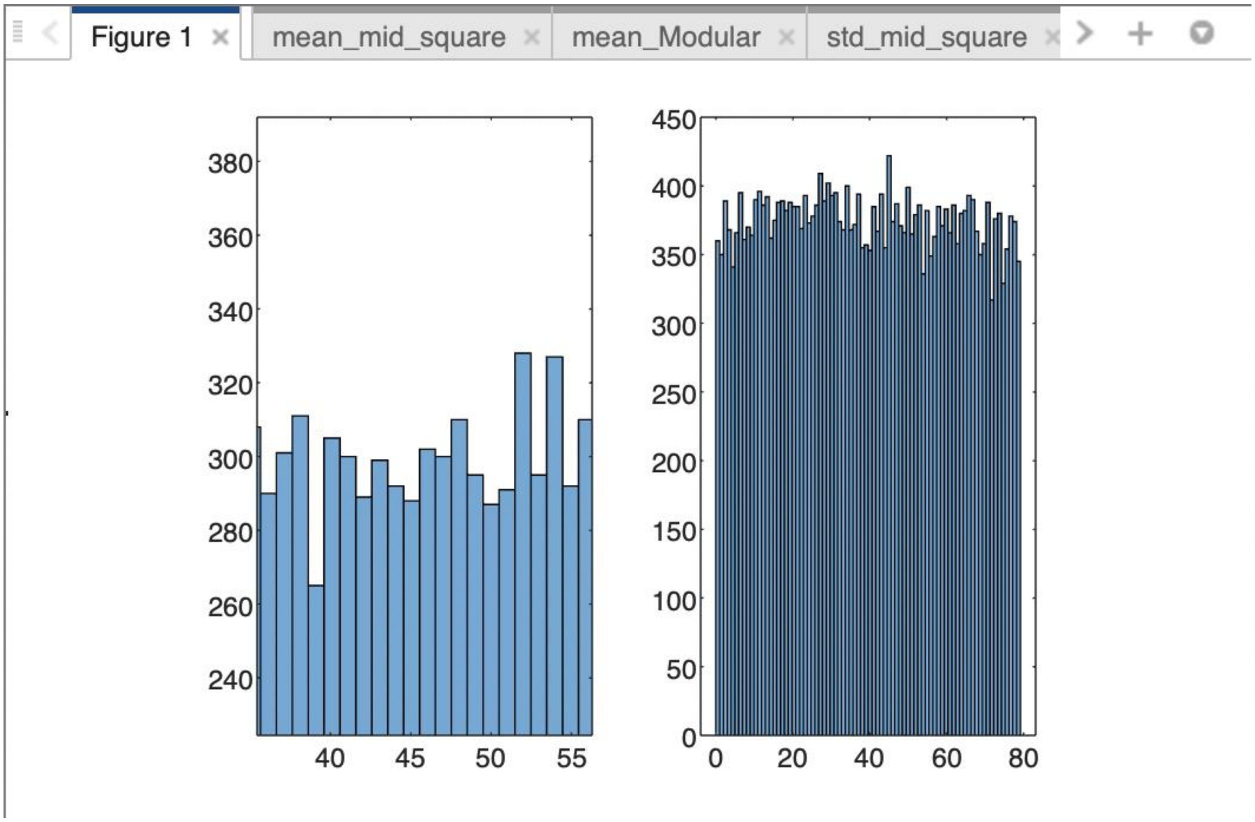
d) is done in code

f) Modular Arithmetic:

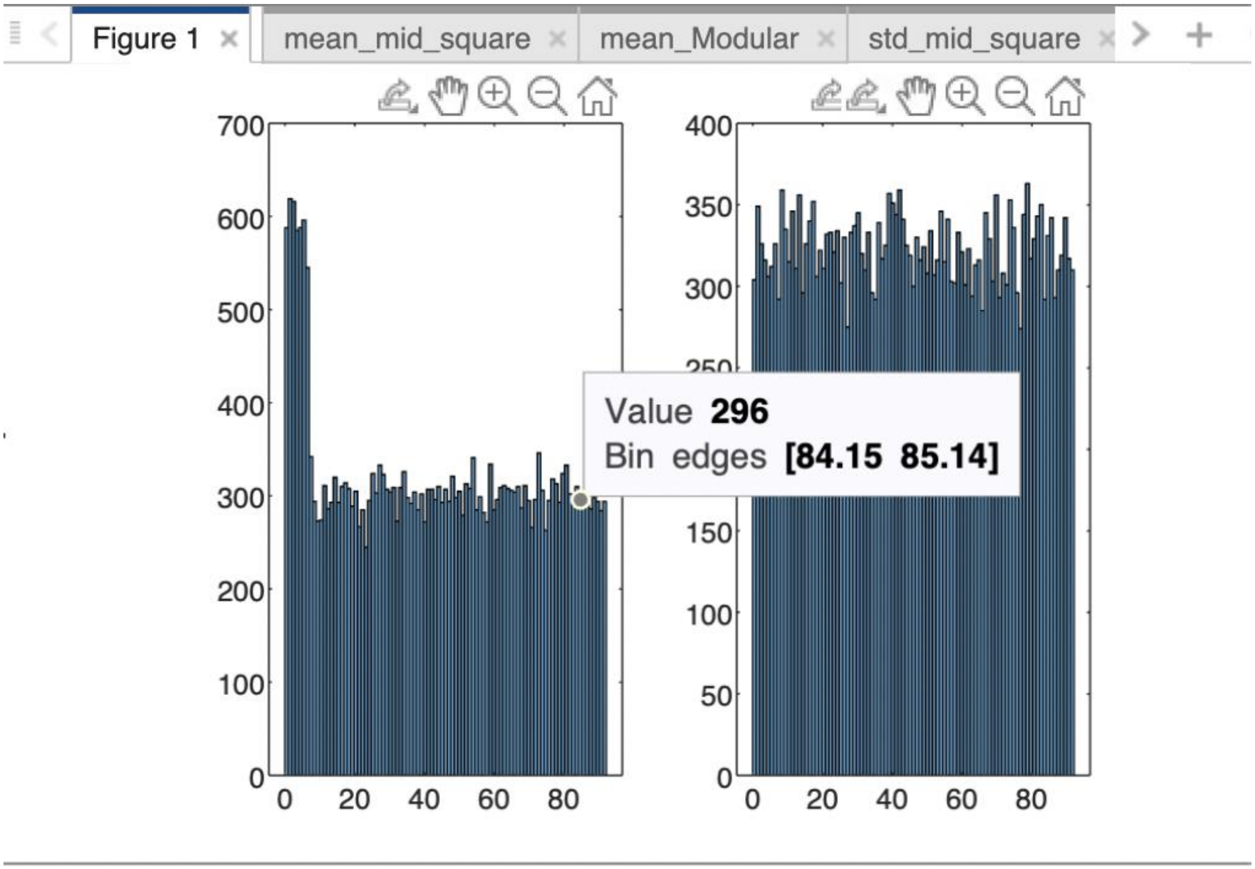
is a function that converts a big input value to a smaller output value using the modulo operator. The fundamental concept is to start with an input value, apply a mathematical operation to it , and then use the modulo operator to decrease the result into a smaller value that may be used as an index in a hash table. (Preneel, 1970)

(b, e, g)

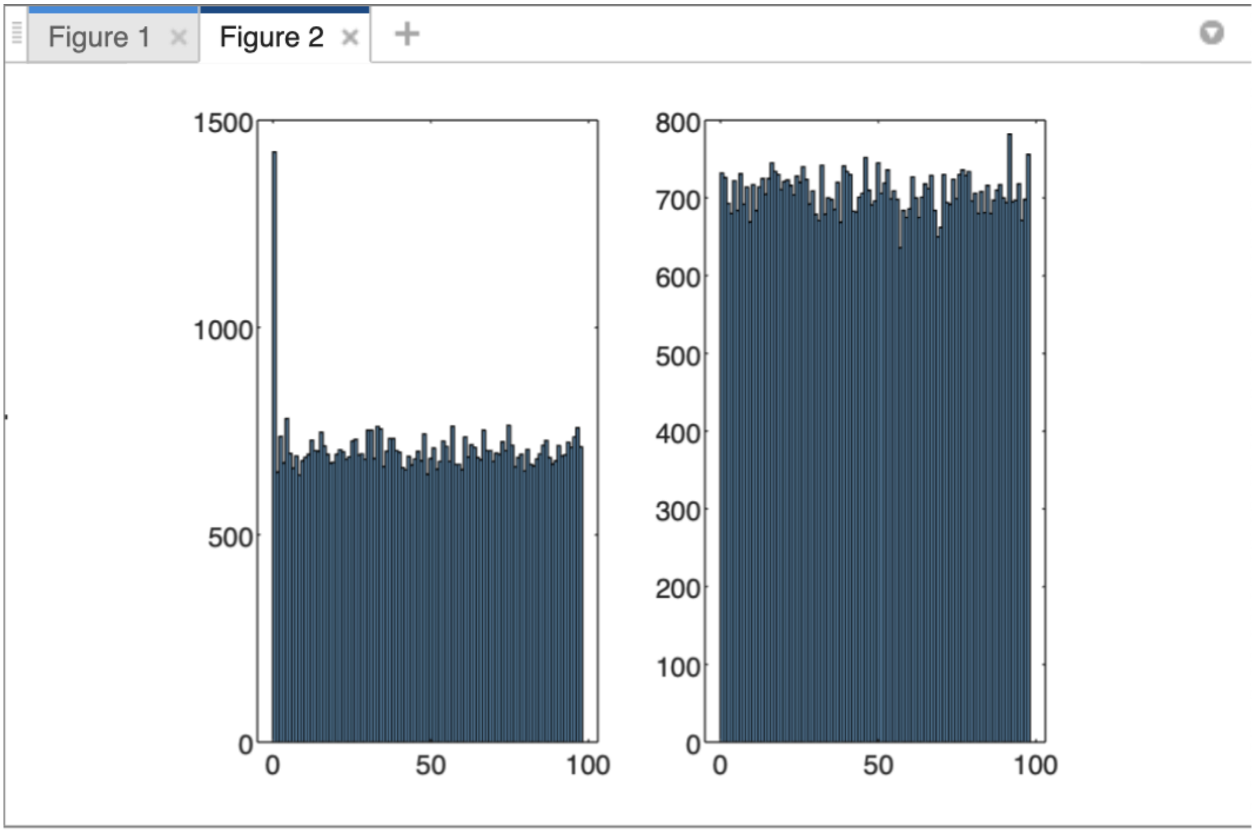
	Number of servers	**Number of requests**	**Standard deviation**	**Mean**
Mid-square method	80	30000	23.958987663876943	33.315143838127938
Modular Arithmetic	80	30000	22.9324188	39.2620316



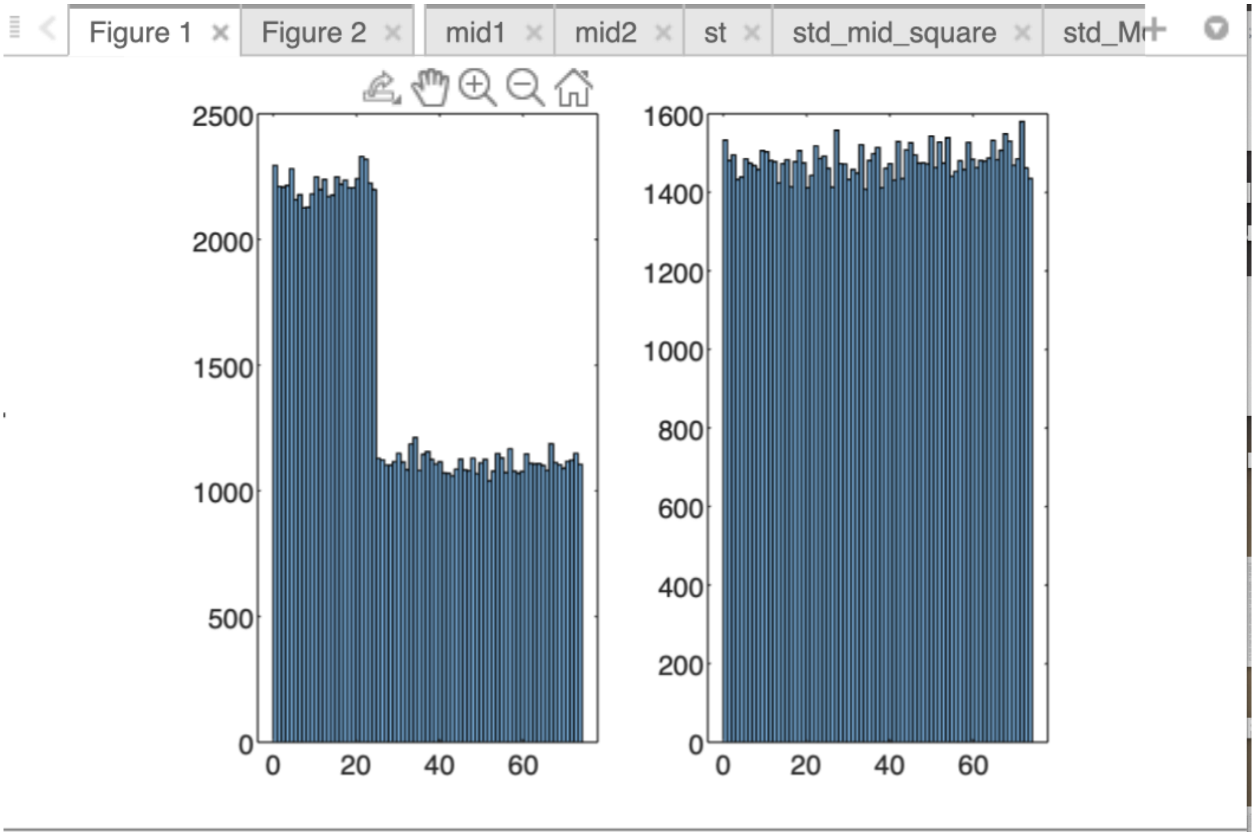
	Number of servers	Number of requests	Standard deviation	mean
Mid-square method	93	30000	28.076211827686343	43.127337577919263
Modular Arithmetic	93	30000	26.8388634	45.8882675



	Number of servers	Number of requests	Standard deviation	mean
Mid-square method	99	70000	28.883182343074594	48.528950413577334
Modular Arithmetic	99	70000	28.6210289	48.9221992



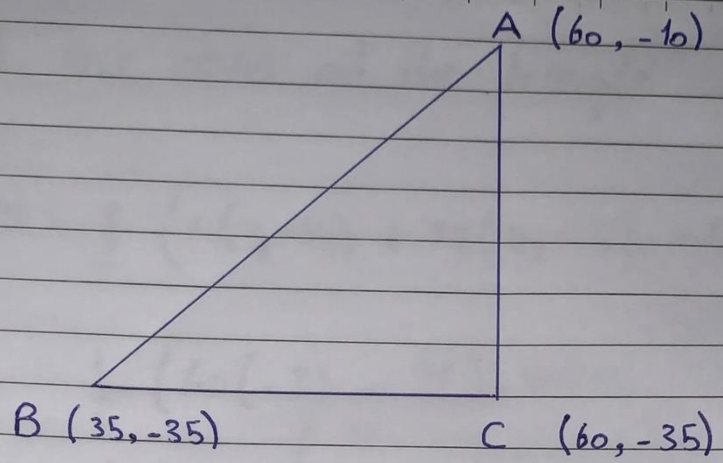
	Number of servers	Number of requests	Standard deviation	Mean
Mid-square method	75	111000	21.929031751327631	15
Modular Arithmetic	75	111000	21.6373711	16



h)

- as we can see in the four graphs, I have done four tries every try was with a different number of requests or different numbers of servers, so we can see the difference between the two functions in the graphs that in all the graphs of the mid-square method that the requests were not equally distributed on the servers but in the modular arithmetic graphs we can see that the requests were almost equally distributed which we can see from this that the modular arithmetic is so much better, also if we look to the mean we can find that the modular arithmetic has a bigger mean than the mid square in all the graphs so this the second point why the modular arithmetic is better, also if we look to the standard deviation we can see that the modular arithmetic standard deviation is less than the mid-square standard deviation with this is the third point why the modular arithmetic is better

Task 4



A) Describe the tringle $\Delta(ABC)$ by vector coordinats

$$\overrightarrow{BA} = \begin{bmatrix} x2 - x_1 \\ y_2 - y_1 \end{bmatrix} = \begin{bmatrix} 60 - 35 \\ -10 - -35 \end{bmatrix} = \begin{bmatrix} 25 \\ 25 \end{bmatrix}$$

$$\overrightarrow{BC} = \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \end{bmatrix} = \begin{bmatrix} 60 - 35 \\ -35 - -35 \end{bmatrix} = \begin{bmatrix} 25 \\ 0 \end{bmatrix}$$

B) Find the area of the tringle $\Delta(ABC)$

area $= \frac{1}{2}\left(x_1(y_2-y_3) + x_2(y_3-y_1) + x_3(y_1-y_2)\right)$

$\frac{1}{2}\left(60(-35--35) + 35(-35--10) + 60(-10--35)\right)$

$(0 \quad -875 \quad 1500)$

$\frac{1}{1}$

$\frac{1}{2}(-875+1500)$

$\frac{1}{2}(625) = 312{,}5$

area

C) Find the angle of the tringle Δ (ABC

$$V_1 \cdot V_2 = V_1x_1 * V_2x_2 + V_1y_1 * V_2y_2$$

$$\overrightarrow{BA} = \begin{bmatrix}25\\25\end{bmatrix} \qquad \overrightarrow{BC} = \begin{bmatrix}25\\0\end{bmatrix}$$

$$25 \times 25 + 25 \times 0 = 625$$

$$||v_1|| = \sqrt{(v_1x_1)^2 + (v_1y_1)^2} = \sqrt{(25)^2 + (25)^2} = 25\sqrt{2}$$

$$||v_2|| = \sqrt{(v_2x_2)^2 + (v_2y_2)^2} = \sqrt{(25)^2 + (0)^2} = 25$$

$$||v_1|| \times ||v_2|| \times \cos\theta = V_1 \cdot V_2$$

$$\frac{25\sqrt{2} \times 25 \times \cos\theta}{25\sqrt{2} \times 25} = \frac{625}{25\sqrt{2} \times 25}$$

$$\cos\theta^{-1} = \frac{625}{25\sqrt{2} \times 25} = \underline{\underline{45}}$$

Angle B

D) $\vec{A} = \begin{bmatrix}60\\-10\end{bmatrix}$ $B = \begin{bmatrix}35\\-35\end{bmatrix}$ $C = \begin{bmatrix}60\\-35\end{bmatrix}$

B must be fixed point

$$T = \begin{bmatrix} Bx + Sx\ (Tx - Bx) \\ By + Sy\ (Ty - By) \end{bmatrix}$$

$$35 + 3\ (Tx - 35)$$

$$-35 + 3\ (Tx - 35)$$

new

$Ax = 35 + 3(60 - 35) = 110$ $A = \begin{bmatrix}110\\40\end{bmatrix}$

$Ay = -35 + 3(-10 + 35) = 40$

new

~~A~~ $Cx = 35 + 3(60 - 35) = 110$

$Cy = -35 + 3(-35 + 35 = -35$

$$C = \begin{bmatrix}110\\-35\end{bmatrix}$$

Fixed point
✓

$$Bx = 35 + 3\underset{0}{\cancel{(35-35)}} = 35$$

$$By = -35 + 3\underset{0}{\cancel{(-35+35)}} = -35$$

stays the same because it is fixed point

STUDENT ASSESSMENT SUBMISSION AND DECLARATION

When submitting evidence for assessment, each student must sign a declaration confirming that the work is their own.

Student name:	Saif Kamal Salim Haddad	**Assessor name:** Dr.Aladeen Al Basheer Dr. Rola Musleh Dr. Hala Hamadeh
Student ID:	21110214	
Issue date: 8/1/2023	**Submission date:** January 29, 2023	**Submitted on:** 27/1/2023
Program: Computing		
HTU Course Name: Maths for Computing	**BTEC UNIT Title** *: Maths for Computing	
HTU Course Code: 40303121	**BTEC UNIT Code:** R/618/7421	
I AM REPEATING THIS UNIT*:	**(YES)** **(NO)**	

Plagiarism

Plagiarism is a particular form of cheating. Plagiarism must be avoided at all costs and students who break the rules, however innocently, may be penalised. It is your responsibility to ensure that you understand **correct referencing practices**. As a university level student, you are expected to use appropriate references throughout andkeep carefully detailed notes of all your sources of materials for material you have used in your work, including any material downloaded from the Internet. Please consult the relevant unit lecturer or your course tutor if you need any further advice.

Student declaration

I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice.

Student signature: Saif Haddad

Date: 27/1/2023

Marijan, B. (2022) *What is load balancing and how does it work?*, *Knowledge Base by phoenixNAP*. Available at: https://phoenixnap.com/kb/load-balancing (Accessed: January 27, 2023).

Smith, B. (2018) *This is how prime numbers keep your online shopping secure*, *ABC News*. ABC News. Available at: https://www.abc.net.au/news/science/2018-01-20/how-prime-numbers-rsa-encryption-works/9338876 (Accessed: January 27, 2023).

GeeksforGeeks (2018) *Mid-square hashing*, *GeeksforGeeks*. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/mid-square-hashing/ (Accessed: January 27, 2023).

Preneel, B. (1970) *MASH hash functions (modular arithmetic secure hash)*, *SpringerLink*. Springer US. Available at: https://link.springer.com/referenceworkentry/10.1007/0-387-23483-7_243#:~:text=MASH%2D1%20and%20MASH%2D2,short%20fixed%20length%20output%20strings. (Accessed: January 27, 2023).