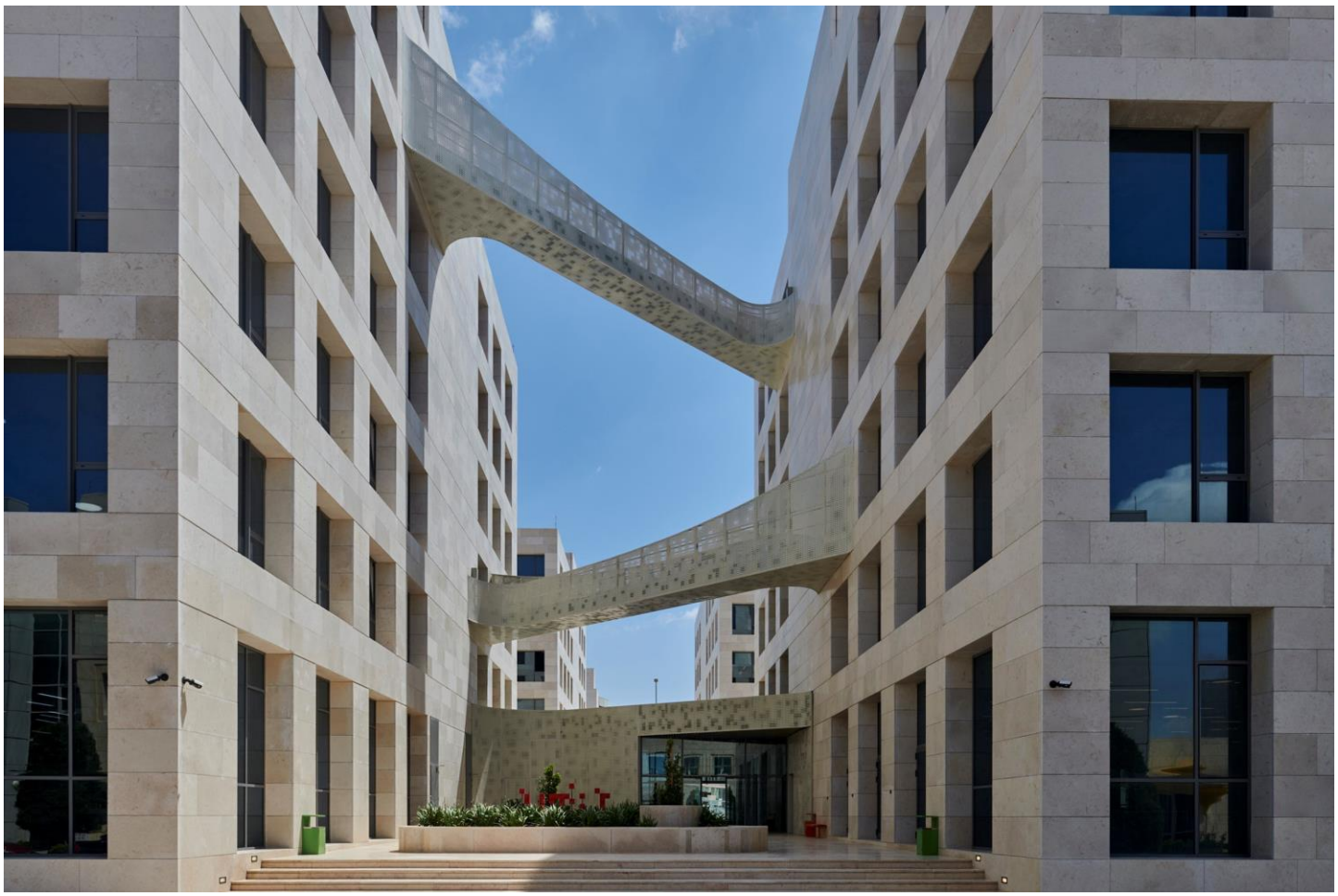
جامعة الحسين التقنية
Al Hussein Technical University

ASSIGNMENT BRIEF

HTU Course No: 40303121	**HTU Course Name:** Maths for Computing
BTEC Unit Code: R/618/7421	**BTEC UNIT Name:** Maths for Computing

Version: 3



Student Name/ID Number/Section	
HTU Course Number and Title	40303121 Maths for Computing
BTEC Unit Code and Title	R/618/7421 Maths for Computing
Academic Year	2023-2024 Spring
Assignment Author	Nehma Al-Tawil
Course Tutor	Nehma Al-Tawil - Rami Hamad - Radwan AlSmadi - Tareq Dalgamoni
Assignment Title	Applications of Mathematical Concepts in the Field of Computing
Assignment Ref No	1
Issue Date	22/04/2024
Formative Assessment dates	From 29/04/2024 to 30/05/2024
Submission Date	09/06/2024
IV Name & Date	Rola Musleh 21/04/2024

Submission Format

In this course, we have one assignment (summative assessment) that is divided into two parts:
Part 1: This is a take-home exam, where some extended calculations are needed and an access toMATLAB software is useful.
Part 2: This is an in-class time constrained exam (i.e., a traditional paper and pencil exam).The dates and times for them are non-negotiable.

Assignment Activities
You must complete both parts and submit them according to the following instructions. Failing to followthese instructions may result in receiving a grade of 'U' in the course.
Take-home Activities:
Criteria covered in this part: M1, M3, D1 and D2. The rest of the criteria in the course is covered in the in-class part.
This part is due on the same day where the in-class part is takes place.Late submissions will not be accepted.

You are required to provide complete answers to all tasks. Use tables or diagrams to support youranswers when necessary.
• Answers need to be provided following clear and coherent steps, using the correct formulas andequations. Final answers without detailed steps are not accepted.
• If a schematic, diagram, or photo is copied, the source must be referenced properly. Quotations arenot allowed by any means. Only re-wording with proper referencing is accepted.
• This is a strictly individual assignment and no collaboration amongst students is allowed, where working with your colleagues is not a teamwork, it is plagiarism. Also, if there was any suspicious then a selective oral will be done.
• The student declaration form attached to this assignment brief must be signed and sent with yourwork (use electronic signature).
• You are required to upload your submission files (source files and MS word file (converted to PDF)) to the university's eLearning system through (https://elearning.htu.edu.jo/) within the submission date and time stated above. NO SUBMISSION by EMAIL and NO LATE SUBMISSIONS

WILL BE ACCEPTED.

Unit Learning Outcomes

LO1 Use applied number theory in practical computing scenarios.

LO2 Analyse events using probability theory and probability distributions.

LO3 Determine solutions of graphical examples using geometry and vector methods.

LO4 Evaluate problems concerning differential and integral calculus.

Assignment Brief and Guidance

You have applied for your dream job at a company that provides computing solutions to its clients. After reviewing your application, the hiring committee found that you are potentially a good match for the position.However, they decided to give you the following tasks to determine whether they should invite you for a personal interview. Do your best to convince the committee that you are the most qualified candidate for this position

Learning Outcomes and Assessment Criteria			
Learning Outcome	**Pass**	**Merit**	**Distinction**
LO1 Use applied number theory in practical computing scenarios.	**P1** Calculate the greatest common divisor and leastcommon multiple of a given pair of numbers. **P2** Use relevant theory tosum arithmetic and geometric progressions.	**M1** Identify multiplicative inverses in modular arithmetic.	**D1** Produce a detailed written explanation of theimportance of prime numbers in the field of computing.
LO2 Analyse events using probability theory and probability distributions.	**P3** Deduce the conditional probability of different events occurring in independent trials. **P4** Identify the expectation of an event occurring from a discrete,random variable.	**M2** Calculate probabilities in both binomially distributed and normally distributed random variables.	**D2** Evaluate probability theory to an example involving hashing and load balancing.
LO3 Determine solutions of graphical examples using geometry and vectormethods.	**P5** Identify simple shapes using co-ordinate geometry. **P6** Determine shapeparameters using appropriate vector methods.	**M3** Evaluate the co-ordinate system used inprogramming a simple output device.	**D3** Construct the scaling of simple shapes that aredescribed by vector co- ordinates.
LO4 Evaluate problems concerning differential and integral calculus.	**P7** Determine the rate ofchange in an algebraic function. **P8** Use integral calculus to solve practical problems involving area.	**M4** Analyse maxima and minima of increasing and decreasing functions, using higher order derivatives.	**D4** Justify, by further differentiation, that avalue is a minimum.

1. Our goal in this task is to encrypt a short message using the RSA algorithm following the steps below.[1]

   - **Step (I): Key Generation**[2]

   In the RSA algorithm, we define

   $$\text{the public key as } PU = \{e, n\}, \text{ and}$$
   $$\text{the private key as } PR = \{d, n\},$$

   where d, e, and n are positive integers. We will obtain these integers by adhering to the following steps.

   (a) Select two primes p and q so that
   - i. p is a three-digit factorial prime. Show that the number you picked is indeed a factorial prime.[3]
   - ii. q is a four-digit Mersenne prime. Show that the number you picked is indeed a Mersenne prime.

   Remark: Here, you need to show first that the chosen number is a prime.

   (b) Compute $n = p \cdot q$ (the system modulus) and $\varphi(n)$ (the Euler's Phi-Function of n).

   (c) Write the prime-power factorization of $\varphi(n)$. What are the prime factors of $\varphi(n)$?

   (d) Choose e to be a Pythagorean prime satisfying the inequality $200 < e < 240$. Show that the number you picked is indeed a Pythagorean prime.

   (e) Explain, from part (c), why e and $\varphi(n)$ are relatively prime. Hence, the multiplicative inverse of e in $\mathbb{Z}_{\varphi(n)}$ exists.

   (f) Showing ***step-by-step details***, calculate d, the multiplicative inverse of e in $\mathbb{Z}_{\varphi(n)}$.

   (g) Summarize the integers you have obtained above in the following table.

p	q	n	$\varphi(n)$	e	d

[1] **Note:** You will find the MATLAB code named "RSA" useful to finish this task.

[2] You can use the MATLAB code to check your work. However, you need to *show all steps* of your work for this step.

[3] See Appendix I for the definition and examples of factorial and Pythagorean primes.

- **Step (II): The RSA Encryption Process**

To encrypt a message, M (in the form of an integer), the sender needs to compute

$$C = M^e \pmod{n},$$

where, $0 \leq M < n$ and C (again, an integer) is the encrypted value of M.

To convert a plaintext message into a positive integer, M, we will use Table 1 (see Appendix II) that shows how some ASCII characters are represented numerically. Using this table and the MATLAB code provided, encrypt the message

I love ”RSA”!

Summarize the results in the following table.

Character	ASCII Code	Encrypted ASCII Code
I		
[space]		
l		
o		
v		
e		
[space]		
”		
R		
S		
A		
”		
!		

Appendix I – Factorial and Pythagorean Prime Numbers

Definition 1. A **factorial prime number** is a prime number that is one less than or one more than a factorial of some positive integer.

Example 1.

(a) 7 is a factorial prime number because 7 is a prime and

$$7 = 6 + 1 = 3! + 1.$$

(b) 23 is a factorial prime number because 23 is a prime and

$$23 = 24 - 1 = 4! - 1.$$

(c) 29 is a prime which is not a factorial prime because there is no integer n so that

$$n! - 1 = 29 \quad \text{or} \quad n! + 1 = 29.$$

That is; the equations

$$n! = 30 \quad \text{or} \quad n! = 28,$$

have no solutions for any positive integer n.

Definition 2. A **Pythagorean prime** is a prime number that can be written in the form $4n + 1$ for some positive integer n.

Example 2.

(a) 5, 13, and 17 are Pythagorean primes because they are primes and

$$5 = 4(1) + 1,$$
$$13 = 4(3) + 1, \text{ and}$$
$$17 = 4(4) + 1.$$

(b) 103 is a prime which is not a Pythagorean prime because there is no integer n so that

$$4n + 1 = 103.$$

That is; the equation

$$4n = 102$$

has no solutions for any positive integer n.

Appendix II – Numerical Representation for Selected ASCII Characters

Code	Character	Code	Character	Code	Character	Code	Character
32	[space]	56	8	80	P	104	h
33	!	57	9	81	Q	105	i
34	”	58	:	82	R	106	j
35	#	59	;	83	S	107	k
36	$	60	<	84	T	108	l
37	%	61	=	85	U	109	m
38	&	62	>	86	V	110	n
39	’	63	?	87	W	111	o
40	(	64	@	88	X	112	p
41	)	65	A	89	Y	113	q
42	*	66	B	90	Z	114	r
43	+	67	C	91	[	115	s
44	’	68	D	92	\	116	t
45	-	69	E	93	]	117	u
46	.	70	F	94	^	118	v
47	/	71	G	95	_	119	w
48	0	72	H	96	‘	120	x
49	1	73	I	97	a	121	y
50	2	74	J	98	b	122	z
51	3	75	K	99	c	123	{
52	4	76	L	100	d	124	|
53	5	77	M	101	e	125	}
54	6	78	N	102	f	126	~
55	7	79	O	103	g	127	[backspace]

Table 1: Numerical representations for selected ASCII characters.

- **Online MATLAB Compiler**

If you do not want to install MATLAB on your computer, you can use the following online compiler: https://www.mathworks.com/products/matlab-online.html to run the code.

2. Consider a scenario where a hash function is used to distribute incoming requests across a set of servers. However, the current implementations of the hash functions, including the Mid-square method and Modular Arithmetic method, lead to a non-uniform distribution of requests among the servers. Your task is to modify the MATLAB code (Hash_Function) for both methods to achieve a more uniform distribution of requests across the servers and compare their effectiveness.

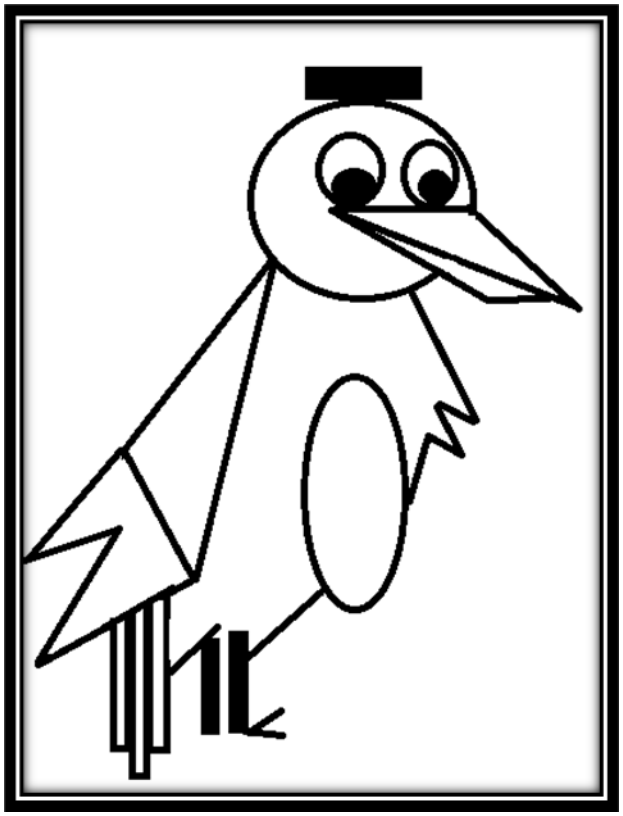
Your tasks are as follows:

a) Analyze the provided MATLAB code and identify the implementations of the Mid-square method and Modular Arithmetic method hash functions.
b) Describe why the current implementations may lead to a non-uniform distribution of requests across servers for each method.
c) Modify both hash functions to achieve a more uniform distribution of requests across the servers.
d) Explain your modifications for each method and discuss how they contribute to achieving a more uniform distribution.
e) Run the modified MATLAB code and plot the distribution of requests across servers for both methods to demonstrate the effectiveness of your modifications.
f) Compare and contrast the distributions obtained from the Mid-square method and Modular Arithmetic method, discussing their respective strengths and weaknesses in achieving uniform distribution.

3. Consider the shape that is given below.

A. Lebel any points (i.e., state their Cartesian Co-ordinates) and use any mathematical equations/conditions that are necessary to produce the given shape by hand. *You may choose your own dimensions/scaling*.

B. Consider the C code named 'Shapes1_Online C compiler'. Modify this code so it produces the given shape using the cartesian coordinates you chose in part **'A'.**



Online C compiler:
https://www.onlinegdb.com/online_c_compiler