HKUST – Department of Computer Science and Engineering
COMP 2711: Discrete Math Tools for Computer Science

Fall 2021 Final Examination

Date: 17 December 2021 Time: 16:30–19:30

Name: _____	Student ID: _____
Email: _____	Lecture: _____

Instructions

- This is a closed book exam. It consists of **20** pages and **8** questions. You are allowed to bring a one-page cheat sheet, handwritten on both sides.
- Please write your name, student ID, email, lecture section in the space provided at the top of this page.
- Please sign the honor code statement on page 2.
- Answer all the questions within the space provided on the examination paper. You may also use the back of the pages. Each question is on a separate page. This is for clarity and is not meant to imply that each question requires a full-page answer. Some can be answered using only a few lines.
- Solutions can be written in terms of binomial coefficients, factorials, and the $C(n,k)$, $P(n,k)$ notation. For example, you can write $\binom{5}{3}+\binom{4}{2}$ instead of 16. Avoid use of nonstandard notation such as ${}_nP_k$ or ${}_nC_k$. Calculators may be used for the exam (but are not necessary).

Questions	1	2	3	4	5	6	7	8	Total
Points	10	10	10	10	10	10	10	10	80
Score									

As part of HKUST's introduction of an honor code, the HKUST Senate has recommended that all students be asked to sign a brief declaration printed on examination answer books that their answers are their own work, and that they are aware of the regulations relating to academic integrity. Following this, please read and sign the declaration below.

I declare that the answers submitted for this examination are my own work.

I understand that sanctions will be imposed, if I am found to have violated the University regulations governing academic integrity.

Student's Name: _____

Student's Signature: _____

Problem 1: Logic [10 pts]

(a) (4 pts) Let $P(x, y)$ represent "$x + y$ is even". Prove or disprove each of the following statements. The domain is all integers.

(i) $\big(\forall x \exists y P(x, y)\big) \rightarrow \big(\exists y \forall x P(x, y)\big)$

(ii) $\Big(\forall x \exists y \big(P(x, y) \wedge \forall z(P(x, z) \rightarrow y = z)\big)\Big) \rightarrow \Big(\exists y \forall x \big(P(x, y) \wedge \forall z(P(x, z) \rightarrow y = z)\big)\Big)$

(b) (4 pts)

(i) Prove or disprove that $\forall x \exists y \exists z(x = yz)$. The domain is all even integers.

(ii) Write a statement that is logically equivalent to the statement in (b) (i) but the domain is all integers. You may use the predicate $E(x)$ to represent "x is even". No other domains and predicates should be defined. Steps/Justifications are not required.

(c) (2 pts) Express the negation of the statement in (a) (ii) so that all negation symbols immediately precede predicates. Steps are not required.

Solution: (a) (i) False. $\big(\forall x \exists y P(x, y)\big)$ is true because we can set $y = x$. $\big(\exists y \forall x P(x, y)\big)$ is false because there exists $x = y + 1$. So, $T \rightarrow F$ is false.

(ii) True. $\Big(\forall x \exists y \big(P(x, y) \wedge \forall z(P(x, z) \rightarrow y = z)\big)\Big)$ is false because there is no unique y to make $P(x, y)$ true, e.g. setting $y = x, y = x + 2, y = x + 4$, etc, can make $P(x, y)$ true. So, the answer is true.

(b) (i) False. 10 is not a product of two even integers.

(ii) $\forall x \Big(E(x) \rightarrow \exists y \Big(E(y) \wedge \exists z \big(E(z) \wedge (x = yz)\big)\Big)\Big)$

(c)

$$
\begin{aligned}
& \neg\Big(\Big(\forall x\exists y\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\to\Big(\exists y\forall x\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\Big)\\
\equiv\ & \neg\Big(\neg\Big(\forall x\exists y\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\vee\Big(\exists y\forall x\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\Big)\\
\equiv\ & \Big(\forall x\exists y\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\wedge\neg\Big(\exists y\forall x\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\\
\equiv\ & \Big(\forall x\exists y\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\wedge\Big(\forall y\exists x\neg\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\\
\equiv\ & \Big(\forall x\exists y\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\wedge\Big(\forall y\exists x\big(\neg P(x,y)\vee\neg\forall z(P(x,z)\to y=z)\big)\Big)\\
\equiv\ & \Big(\forall x\exists y\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\wedge\Big(\forall y\exists x\big(\neg P(x,y)\vee\exists z\neg(P(x,z)\to y=z)\big)\Big)\\
\equiv\ & \Big(\forall x\exists y\big(P(x,y)\wedge\forall z(P(x,z)\to y=z)\big)\Big)\wedge\Big(\forall y\exists x\big(\neg P(x,y)\vee\exists z(P(x,z)\wedge y\neq z)\big)\Big)
\end{aligned}
$$

Problem 2: Counting [10 pts]

(a) (2.5 pts) How many ways are there to distribute seven different objects into four distinct containers with no container left empty?

(b) (2.5 pts) Now suppose the containers are identical in appearance. How many ways are there to distribute the seven different objects into these four identical containers, with no container left empty?

(c) (5 pts) Let $S(m, n)$ denote the number of ways in which it is possible to distribute m distinct objects into n identical containers, with no container left empty. Give a combinatorial proof for the following identity

$$S(m+1, n) = S(m, n-1) + nS(m, n)$$

Solution: (a) $\binom{4}{4}4^7 - \binom{4}{3}3^7 + \binom{4}{2}2^7 - \binom{4}{1}1^7 = 8400$

(b) 8400/4!

(c) Let $A = \{a_1, a_2...., a_m, a_{m+1}\}$. Left hand side counts the number of ways in which the objects of A can be distributed among n identical containers, with no container left empty. Right-hand side applies the sum rule to counting two types of distributions: type 1 where a_{m+1} is in a container by itself and type 2 where a_{m+1} is in the same container as another object from A. For type 1 distributions, we distribute $a_1, a_2, ..., a_m$ among $n-1$ identifical containers, with none left empty. Then place a_{m+1} in the remaining empty container. This results in $S(m, n-1)$ ways. For type 2 distributions, we distribute $a_1, a_2, ..., a_m$ among the n identical containers with none left empty. For each of these $S(m, n)$ ways, the n containers become distinguishable by their contents. Selecting one of the n distinct containers for a_{m+1} leads to a total of $nS(m, n)$ type 2 distributions.

Problem 3: Number Theory [10 pts]

Solve each of the congruences with $0 \leq x, y \leq 111$. You only need to find one value of x and/or y even though there may be multiple solutions in the given range. You may use the results $25 \cdot 9 \equiv 1 \pmod{112}$, $8 \cdot 1 \equiv 1 \pmod{7}$ and $7 \cdot 7 \equiv 1 \pmod{8}$. Show all steps.

(a) (3 pts)

$$5x - 41 \equiv 8 \pmod{112}$$

(b) (3 pts)

$$\begin{cases} 21x + 17y \equiv 19 \pmod{112} \\ 3x + 2y \equiv 2 \pmod{112} \end{cases}$$

(c) (4 pts)

$$\begin{cases} x \equiv 11 \pmod{14} \\ x \equiv 5 \pmod{8} \end{cases}$$

Solution: (a)

$$\begin{aligned} & 5x - 41 \equiv 8 \pmod{112} \\ \Rightarrow \quad & 5x \equiv 49 \pmod{112} \\ \Rightarrow \quad & 9 \cdot 5 \cdot 5x \equiv 45(49) \pmod{112} \\ \Rightarrow \quad & x \equiv 45(49) \pmod{112} \\ \Rightarrow \quad & x \equiv 77 \pmod{112} \end{aligned}$$

(b)

$$\begin{aligned} & \begin{cases} 21x + 17y \equiv 19 \pmod{112} \\ (-7)(3x + 2y) \equiv (-7) \cdot 2 \pmod{112} \end{cases} \\ \Rightarrow \quad & 21x + 17y - 21x - 14y \equiv 19 - 14 \pmod{112} \\ \Rightarrow \quad & 3y \equiv 5 \pmod{112} \\ \Rightarrow \quad & 25 \cdot 3 \cdot 3y \equiv 75 \cdot 5 \pmod{112} \\ \Rightarrow \quad & y \equiv 39 \pmod{112} \end{aligned}$$

When $y = 39$, we have $3x + 78 \equiv 2 \pmod{112}$, and thus $x \equiv (75)(-76) \equiv 12 \pmod{112}$.

(c) This system is equivalent to

$$\begin{cases} x \equiv 11 \equiv 4 \pmod{7} \\ x \equiv 5 \pmod{8} \end{cases}$$

7 and 8 are relatively prime. By the Chinese remainder theorem, $x \equiv 109 \equiv 53 \pmod{56}$.

Problem 4: Induction [10 pts]

Prove the following using different variants of induction:

(i) (3 pts) You are given infinitely many coins of 2 and 5 cents. Use regular (not strong) induction to prove that you can create any integer amount $n \geq 4$.

(ii) (3 pts) You are given a rectangular chocolate bar of dimensions n by m that consists of $n \cdot m$ same-size squares, separated by horizontal and vertical lines. You want to break it into $n \cdot m$ separate square pieces. Assume that you can only "cut" along an entire horizontal or vertical line each time. Use strong induction on the size of the bar to prove that in order to break it into $n \cdot m$ square pieces you need exactly $n \cdot m - 1$ "cuts".

(iii) (4 pts) Given a string x, its *reversal* $\mathsf{Rev}(x)$ is the string that has the same characters but in the reverse order. E.g., for the string "*abcd*" its reversal is "*dcba*". This can be defined recursively as:

$$\mathsf{Rev}(x) = \begin{cases} \lambda & \text{if } x = \lambda \text{ where } \lambda \text{ is the empty string} \\ z\mathsf{Rev}(y) & \text{if } x = yz \text{ for a single character } z \text{ appended to another string } y \end{cases}$$

Use structural induction to prove that $\mathsf{Rev}(x_1x_2) = \mathsf{Rev}(x_2)\mathsf{Rev}(x_1)$. This means that the reversal of a string consisting of the concatenation of strings x_1 and x_2 is equal to the concatenation of the reversal of x_2 and the reversal of x_1. *(Hint: try structural induction for all x_2 created using the above recursive definition. That is, show that the proposition holds for all possible x_2 starting from the simplest case.)*

Solution: (i) Base step: $n = 4$ can be created using two 2-cent coins.

Inductive step: Assuming k can be created by 2 and 5-cent coins, we show this is also true for $k + 1$. First, since $k \geq 4$, it must be true that the coins used for k include at least two 2-cent coins or one 5-cent coin. So we can separate the remaining of the proof into two cases. If the coins for k include a 5-cent coin, remove it and replace it with three 2-cent coins. If the coins for k include two 2-cent coins, remove them and replace them with one 5-cent coin. In both cases, the target $k + 1$ has been achieved.

(ii) Base step: The smallest bar is for $n = m = 1$. In that case, it takes 0 "cuts" to split the bar and $n * m - 1 = 1 * 1 - 1 = 0$.

Inductive step. Assuming the proposition holds for k-sized bars for all $1 \leq j \leq k$, we will show it takes k "cuts" for $k + 1$-size bar. First, make 1 "cut" along any line. The result is two bars of sizes k_1, k_2 with $k_1 + k_2 = k + 1$. Since $1 \leq k_1, k_2 \leq k$, it follows that the first bar can

be further split into k_1 squares with $k_1 - 1$ "cuts" and the second can be further split into k_2 squares with $k_2 - 1$ "cuts". So, in total splitting the original bar takes $1 + k_1 - 1 + k_2 - 1 = 1 + k + 1 - 1 - 1 = k$ splits.

(iii) Base step: If $x_2 = \lambda$ then $\mathsf{Rev}(x_1\lambda) = \mathsf{Rev}(x_1)$. Likewise $\mathsf{Rev}(\lambda)\mathsf{Rev}(x_1) = \lambda\mathsf{Rev}(x_1) = \mathsf{Rev}(x_1)$.

Inductive step: Let $x_2 = x_3 a$ for some character a and a string x_3 with length one less than that of x_2 for which the property holds. Then $\mathsf{Rev}(x_1x_2) = \mathsf{Rev}(x_1x_3a)$. From the recursive definition, $\mathsf{Rev}(x_1x_3a) = a\mathsf{Rev}(x_1x_3)$. Also, from the inductive hypothesis $\mathsf{Rev}(x_1x_3) = \mathsf{Rev}(x_3)\mathsf{Rev}(x_1)$. So we can write $\mathsf{Rev}(x_1x_2) = a\mathsf{Rev}(x_3)\mathsf{Rev}(x_1) = \mathsf{Rev}(x_3a)\mathsf{Rev}(x_1) = \mathsf{Rev}(x_2)\mathsf{Rev}(x_1)$, where we again used the recursive definition.

Problem 5: Algorithm Analysis [10 pts]

You are given a binary array of length n. It begins with a number of 1s, at some position it changes to 0 and it continues with 0s until the end. The array is guaranteed to have at least one 1 and at least one 0.

The problem you have to solve is to find at which position the "change" from 1s to 0s happens (equivalently, what is the position of the last 1 in the array).

(i) (2 pts) Give an algorithm that solves the above problem with $\Theta(n)$ worst-case behavior. Prove it really is $\Theta(n)$.

(ii) (4 pts) Give an algorithm that solves the above problem with $\Theta(\log n)$ worst-case behavior. Prove it really is $\Theta(\log n)$.

(iii) (4 pts) Assume that the array begins with k 1s for some (unknown) $0 < k < n$. Give an algorithm that solves the above problem with $\Theta(\log k)$ worst-case behavior. Prove it really is $\Theta(\log k)$.

You may present your algorithms either in pseudocode form or with a detailed explanation in text.

Solution: (i) Linear search from the beginning till you find the position where the "change" happens. It checks up to $n-1$ locations sequentially so $O(n)$. If there are $n-1$ 1s, it has to go over the entire array so $\Omega(n)$. Hence $\Theta(n)$.

(ii) A modified version of binary search where, at each repetition, the starting position of the search moves up if the middle is 0, or the ending position of the search moves down if the middle is 1. A slightly more efficient version will check two positions at each iteration (middle, middle+1) hoping to find the "change" position faster. Either way, since the search space is initially n and it is halved in every iteration, it performs $O(\log n)$ checks. Depending on the given version of binary search, it always performs this many checks, or it is forced to perform this many checks if the "change" position is found at the last iteration. So $\Omega(\log n)$, hence $\Theta(\log n)$.

iii The algorithm is based on a similar idea as repeated squaring. Set position $i = 1$. While $array[i] == 1$, set $i \leftarrow i * 2$. When you get $array[i] == 0$, run the "binary-search" style algorithm from part (ii) between $array[i/2]$ and $array[i]$ and return its result. Since $array[i/2] = 1$, (as the loop continued doubling i) but $array[i] = 0$, the "change" happened between these two positions. The first part takes $\lfloor \log k \rfloor + 1$ steps as there are k consecutive 1s in the beginning of the array. For the second part, $i - i/2 < k$ so from the analysis of part (ii) this

takes $O(\log k)$ steps. So the algorithm has $O(\log k)$ worst-case runtime. Since the first part always takes $\lfloor \log k \rfloor + 1$ steps, it is also $\Omega(\log k)$ hence $\Theta(\log k)$.

Problem 6: RSA [10 pts]

Alice constructs an RSA key-pair. She first chooses $p = 11, q = 17$ and sets $n = 11 \cdot 17 = 187$. Her decryption key is $d = 7$.

Similarly, Bob constructs an RSA key-pair. He first chooses $p = 7, q = 23$ and sets $n = 7 \cdot 23 = 161$. His public key is $(e, n) = (71, 161)$.

(a) (3 pts) What is Alice's public key?

(b) (3 pts) What is Bob's decryption key d?

(c) (4 pts) Alice wants to send Bob the message $M = 100$. She encrypts the message by the RSA algorithm. What is the value of the encrypted message that Alice sends Bob?

For each part, show all steps of your computation.

Solution: (a) $T = (p-1)(q-1) = 10 \cdot 16 = 160$
The inverse of 7 in Z_{160} is 23.
Thus, Alice's public key is $(e, n) = (23, 187)$.
Note. One way of finding that 23 is the inverse of 7 in Z_{160} is by running the extended GCD algorithm on $7, 160$ to get

$$1 = 7 \cdot 23 - 160.$$

(b) $T = (p-1)(q-1) = 6 \cdot 22 = 132$
The inverse of 71 in Z_{132} is 119.
Thus, Bob's secret key is $d = 119$.
Note. One way of finding that 119 is the inverse of 71 in Z_{132} is by running the extended GCD algorithm on $71, 132$ to get

$$1 = (-13) \cdot 71 - 7 \cdot 132.$$

The inverse of 71 in Z_{132} is $(-13) \bmod 132 = 119$.

(c) Since Alice wants to send a message to Bob, she uses Bob's public key to encrypt the message. Thus, the encrypted message is $M^{71} \bmod 161 = 100^{71} \bmod 161 = 39$.

By repeated squaring:
$100^1 \bmod 161 = 100$.
$100^2 \bmod 161 = 18$.
$100^4 \bmod 161 = 18^2 \bmod 161 = 2$.
$100^8 \bmod 161 = 2^2 \bmod 161 = 4$.
$100^{16} \bmod 161 = 4^2 \bmod 161 = 16$.
$100^{32} \bmod 161 = 16^2 \bmod 161 = 95$.
$100^{64} \bmod 161 = 95^2 \bmod 161 = 9$.

$$100^{71} \bmod 161 = 9 \cdot 2 \cdot 18 \cdot 100 \bmod 161 = 32400 \bmod 161 = 39.$$

Problem 7: Expectation and Variance [10 pts]
Assume there were 3 types of questions in last year's COMP2711 final exam:

- 20 True/False Questions worth 1 point each.
- 1 Medium Question worth 10 points.
- 1 Hard Question worth 50 points.

Assuming that all questions are mutually independent, and the grading scheme of the exam is as follows:

- The True/False Questions are graded by flipping a coin for each question and scoring the question as 1 if the coin lands heads, and 0 otherwise.
- The Medium Question is graded by Alice or Bob. Alice has a 40% chance of grading the question, and chooses a random score uniformly from 0 to 10 inclusive. Bob has a 60% chance of grading the question, and he gives 0 with 70% probability and 10 with 30% probability.
- The Hard Question is graded by Charles, who rolls two 6-sided fair dice, and subtracts their product from 50. E.g. If he rolls 3,4 the score given to that question is 50 - 12 = 38.

(a) (3 pts) What is the expected score of the exam?

(b) (3 pts) What is the variance of the True/False section?

(c) (4 pts) What is the variance of the Medium Question given that Bob is grading the question?

Solution: (a) $E[Score] = E[T/Fscore] + E[MedScore] + E[HardScore]$
$E[T/Fscore] = 20/2 = 10$
$E[MedScore] = 0.4 * 5 + 0.6 * 3 = 3.8$
$E[HardScore] = 50 - 3.5^2 = 37.75$
$E[Score] = 51.55/80 = 64.43\%$

(b) $Var[T/FScore] = 20 * Var[1T/F] = 20 * 0.5 * 0.5 = 5$

(c) If Bob is grading, then the values of X are 0 and 10 with probability 0.7 and 0.3 respectively.
$E[X] = 0.3 * 10 = 3.$
$E[X^2] = 0^2 * 0.7 + 10^2 * 0.3 = 30.$
$Var[MedScore] = E[X^2] - E[X]^2 = 30 - 9 = 21$

Problem 8: Recurrences [10 pts]

(a) (4 pts) Solve the recurrence equation

$$T(0) = 3$$
$$T(n) = 2T(n-3) + 5 \qquad \text{for } n > 0$$

where n is a multiple of 3. Use the direct method and show all steps of your computation. *(Hint: the answer is* $8 \cdot 2^{n/3} - 5$*.)*

(b) (6 pts) Prove your answer in (a) by induction.

Solution:

(a)

$$\begin{aligned}
T(n) &= 2T(n-3) + 5 \\
&= 2(2T(n-2\cdot 3) + 5) + 5 \\
&= 2^2 T(n-2\cdot 3) + 2^1 \cdot 5 + 2^0 \cdot 5 \\
&\vdots \\
&= 2^h T(n-h\cdot 3) + 2^{h-1} \cdot 5 + \cdots + 2^0 \cdot 5 \\
&\quad \text{Set } h = n/3 \\
&= 2^h T(0) + 2^{h-1} \cdot 5 + \cdots + 2^0 \cdot 5 \\
&= 3 \cdot 2^h + 2^{h-1} \cdot 5 + \cdots + 2^0 \cdot 5 \\
&= 3 \cdot 2^{n/3} + 5\frac{2^{n/3}-1}{2-1} \\
&= 3 \cdot 2^{n/3} + 5 \cdot 2^{n/3} - 5 \\
&= 8 \cdot 2^{n/3} - 5
\end{aligned}$$

(b) **Base case**: $T(0) = 3 = 8 \cdot 2^{0/3} - 5$
Inductive hypothesis: Assume $T(n-3) = 8 \cdot 2^{(n-3)/3} - 5$ is true for $n > 0$ and n is multiple of 3.
Inductive step:

$$\begin{aligned}
T(n) &= 2T(n-3) + 5 \\
&= 2(8 \cdot 2^{(n-3)/3} - 5) + 5 \\
&= 8 \cdot 2^{(n-3+3)/3} - 10 + 5 \\
&= 8 \cdot 2^{n/3} - 5
\end{aligned}$$

=== Extra Space ===

=== Extra Space ===