University of Toronto
Faculty of Arts and Science

CSC165H1S APRIL 2019 EXAMINATIONS

Duration: 3 hours **Instructor(s):** David Liu, François Pitt

No Aids Allowed

Name: | | | | | | | | | | | | | | | | | | | | | | | | |

Student Number: | | | | | | | | | | |

- Please write your name and student number on the front of the exam.
- This examination has **9** questions. There are a total of **18 pages, DOUBLE-SIDED**.
- All statements in predicate logic must have negations applied directly to propositional variables or predicates.
- In your proofs, you may always use definitions we have covered in this course. However, you may **not** use any external facts about these definitions unless they are given in the question.
- For algorithm analysis questions, you can jump immediately from an exact step count to an asymptotic bound without proof (e.g., write "the number of steps is $3n + \lceil \log n \rceil$, which is $\Theta(n)$").
- **You must earn a grade of at least 40% on this exam to pass this course.**

As a student, you help create a fair and inclusive writing environment. If you possess an unauthorized aid during an exam, you may be charged with an academic offence. Turn off and place all cell phones, smart watches, electronic devices, and unauthorized study materials in your bag under your desk. If it is left in your pocket, it may be an academic offence. Students must hand in all examination materials at the end.

Take a deep breath.

This is your chance to show us
how much you've learned.
We **WANT** to give you the credit
that you've earned.
A number does not define you.

It's been a real pleasure teaching you this term.

Good luck!

Question	Grade	Out of
Q1		8
Q2		7
Q3		9
Q4		8
Q5		6
Q6		6
Q7		6
Q8		7
Q9		11
Total		68

1. **[8 marks] Translations.** Let P be the set of all people and M the set of all movies. We define these predicates:

   - $Adult(p)$: "p is an adult", where $p \in P$
   - $Scifi(m)$: "m is a science fiction movie", where $m \in M$
   - $Watched(p, m)$: "p has watched m", where $p \in P$ and $m \in M$

   Translate each of the following statements into predicate logic. No explanation is necessary. Do not define any other predicates or sets. You may use "$=$" and "$\neq$" to compare whether two objects are the same.

   (a) **[2 marks]** Every person has watched at least two different science fiction movies.

   (b) **[2 marks]** There is an adult who has watched every movie that is *not* a science fiction movie.

   (c) **[2 marks]** If there is at least one science fiction movie, then every adult has watched at least one science fiction movie.

   (d) **[2 marks]** Every movie has been watched by at most one person.

2. **[7 marks] Divisibility.** Consider the following statement:

   For every natural number n, if $n > 1$ and n is not prime then n is divisible by a number between 2 and $\sqrt{n}$ inclusive.

   (a) **[2 marks]** Translate the above statement into predicate logic. You may use the divisibility predicate "$|$", but may not use the *Prime* predicate (instead, expand it into its definition).

   (b) **[5 marks]** Prove the above statement.

3. **[9 marks] Summing roots.**

(a) **[3 marks]** Prove that for all $n \in \mathbb{N}$, $\sqrt{n+1} - \sqrt{n} > \dfrac{1}{2\sqrt{n+1}}$.

HINT: use difference of squares. You don't need induction here.

(b) **[6 marks]** Consider the following statement.

$$\forall n \in \mathbb{N},\ n \geq 5 \Rightarrow \sum_{i=1}^{n} \sqrt{i} < \frac{3}{4} n\sqrt{n}$$

We want to do a proof by induction of this statement. In the space below, write the **induction step** for this proof. You do *not* need to write a base case.

Use the statement from part (a), as well as this hint: $k\sqrt{k} = (k+1)\sqrt{k+1} - \sqrt{k+1} - k\left(\sqrt{k+1} - \sqrt{k}\right)$.

4. [**8 marks**] **Number representations.** We define the predicate $BT(n,x)$: "x has a balanced ternary representation that contains n digits," where $n \in \mathbb{Z}^+$ and $x \in \mathbb{Z}$. Equivalently,

$$BT(n,x): \ \exists d_0, d_1, \ldots, d_{n-1} \in \{-1,0,1\}, \ \sum_{i=0}^{n-1} d_i \cdot 3^i = x$$

Prove the following statement using induction on n:

$$\forall n \in \mathbb{Z}^+, \ \forall x \in \mathbb{Z}, \ \ -\frac{3^n-1}{2} \leq x \leq \frac{3^n-1}{2} \Rightarrow BT(n,x)$$

The next page is left blank for rough work and/or to continue your proof.

Use this page for rough work. If you want work on this page to be marked, please indicate this clearly *at the location of the original question.*

5. **[6 marks] Asymptotic analysis.** Prove or disprove the following statement. If you are disproving it, begin by writing its negation; you may, but are not required to, expand the definition of Big-Oh in the negated statement.

$$\forall a \in \mathbb{R}^+,\ an^4 + 1 \in \mathcal{O}(n^4 - n)$$

6. **[6 marks] Algorithm running time.** Consider the following two functions.

```
def is_prime(n: int) -> bool:
    """Return whether n is prime. Precondition: n >= 2."""
    for d in range(2, n):          # d = 2, 3, ..., n-1.
        if n % d == 0:
            return False
    return True

def print_primes(n: int) -> None:
    """Print all primes <= n. Precondition: n >= 2."""
    for k in range(2, n + 1):      # k = 2, 3, ..., n
        if is_prime(k):
            print(k)
```

Prove that the running time of print_primes is $\mathcal{O}\left(\frac{n^2}{\log n}\right)$. You may use the statements from Questions 2 and 3 and the following fact, as long as you clearly state where you use each one.

$$\left(\sum_{\substack{2 \le k \le n \\ k \text{ is prime}}} k\right) \in \mathcal{O}\left(\frac{n^2}{\log n}\right) \tag{Fact 1}$$

7. **[6 marks] Running time analysis.** Consider the following algorithm.

```
1  def my_alg(lst: List[int]) -> None:
2      n = len(lst)
3      count = 1
4      for i in range(n):              # Loop 1
5          if lst[i] != 1:
6              count = count * 2
7          else:
8              for j in range(count):  # Loop 2
9                  print(j)
10             return
```

Prove matching upper (Big-Oh) and lower (Omega) bounds on the worst-case running time of my_alg. Clearly label which part of your solution is a proof of the upper bound, and which part is a proof of the lower bound. To simplify your calculation, you may count only the cost of Lines 6 and 9 in your step count, ignoring the running times of the other operations.

The next page is left blank for rough work and/or to continue your proof.

Use this page for rough work. If you want work on this page to be marked, please indicate this clearly *at the location of the original question.*

8. **[7 marks] Average-case analysis.** Here is the same algorithm as in Question 7.

```
1  def my_alg(lst: List[int]) -> None:
2      n = len(lst)
3      count = 1
4      for i in range(n):              # Loop 1
5          if lst[i] != 1:
6              count = count * 2
7          else:
8              for j in range(count):  # Loop 2
9                  print(j)
10             return
```

For this question, we will only consider the following inputs: for each $n \in \mathbb{N}$, we define the set $\mathcal{I}_n$ to be the set of lists of length n that are *permutations* of the numbers $\{1, 2, \ldots, n\}$.

(a) **[2 marks]** Let $n \in \mathbb{N}$, and assume $n \geq 1$. Let $i \in \mathbb{N}$ and assume $0 \leq i < n$. How many inputs in $\mathcal{I}_n$ have a 1 at position i, in terms of n and/or i? Briefly justify your answer.

(b) **[5 marks]** Calculate the average running time of `my_alg` for the set of inputs $\mathcal{I}_n$ defined above. To simplify your calculation, you may count only the cost of Lines 6 and 9 in your step count, ignoring the running times of the other operations. We are looking for both an exact calculation and a Theta expression at the end.
The next page is left blank for rough work and/or to continue your solution.

Use this page for rough work. If you want work on this page to be marked, please indicate this clearly *at the location of the original question.*

9. [11 marks] **Graphs.**

(a) **[2 marks]** Consider the following theorem, which is a variation on a theorem we discussed in the course.

Theorem. For all graphs $G = (V, E)$ and all $u, v, w_1, w_2 \in V$, if these assumptions are true:

- u is connected to w_1 or w_2 in G, and v is connected to w_1 or w_2 in G, and
- w_1 and w_2 are connected in G,

then u and v are also connected in G.

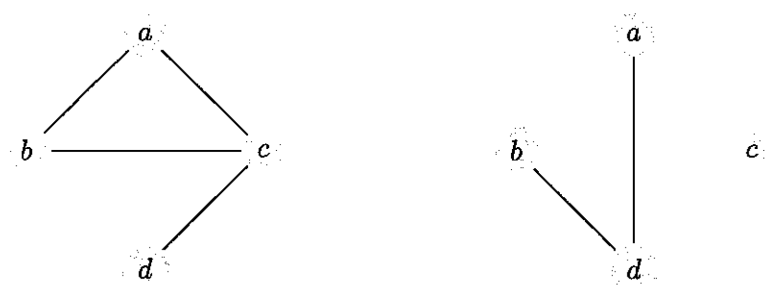
Write the *contrapositive form* of this statement in predicate logic. Use the predicate $Conn(G, u, v)$ to mean "u and v are connected in G" in your translation. Use $\forall G = (V, E)$ to quantify over all graphs.

Consider the following definition.

Definition 1. Let $G = (V, E)$ be a graph. The **complement** of G is the graph $G^c = (V, E^c)$, where:

- The set of vertices V for G^c is the same as for G.
- The set of edges E^c for G^c is defined as $E^c = \{(u, v) \mid u, v \in V \text{ and } (u, v) \notin E\}$.

We have drawn an example of a graph G (on the left) and its complement G^c (on the right) below.



(b) **[2 marks]** Let $G = (V, E)$ be a graph, let $G^c = (V, E^c)$ be its complement, and let $n = |V|$. Write a formula for $|E^c|$ (the number of edges of G^c) in terms of n and/or $|E|$. Briefly justify your answer.

(c) **[2 marks]** Translate the following statement into predicate logic. You may again use the predicate $Conn(G, u, v)$, but you must express what it means for an entire graph itself to be connected in terms of this predicate.

"For every graph $G = (V, E)$, if G is not connected then its complement G^c is connected."

(d) **[5 marks]** Prove the statement from part (c). You may use the theorem from part (a) (original or contrapositive form) as an external fact in your proof.

HINT: pick arbitrary $u, v \in V$, and divide your proof into two cases, when $(u, v) \in E^c$ and when $(u, v) \notin E^c$.

Use this page for rough work. If you want work on this page to be marked, please indicate this clearly *at the location of the original question.*

Use this page for rough work. If you want work on this page to be marked, please indicate this clearly *at the location of the original question*.

Use this page for rough work. If you want work on this page to be marked, please indicate this clearly *at the location of the original question.*

CSC165 April 2019 Examination Aid Sheet

You may use all definitions and formulas on this aid sheet on all parts of the exam.

Standard numeric sets

- $\mathbb{N}$ (natural numbers); $0 \in \mathbb{N}$
- $\mathbb{Z}$ (integers)
- $\mathbb{Z}^+$ (positive integers)
- $\mathbb{R}$ (real numbers)
- $\mathbb{R}^{\geq 0}$ (non-negative reals)
- $\mathbb{R}^+$ (positive reals)

Summations and Products

- For all $n \in \mathbb{N}$ and $r \in \mathbb{R}$ where $r \neq 1$:

$$\sum_{i=0}^{n} i = \frac{n(n+1)}{2}$$

$$\sum_{i=0}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$$

$$\sum_{i=0}^{n} r^i = \frac{r^{n+1}-1}{r-1}$$

$$\sum_{i=0}^{n} i \cdot r^i = \frac{nr^{n+1}}{r-1} - \frac{r(r^n-1)}{(r-1)^2}$$

- For all $m, n \in \mathbb{Z}$, if $m > n$ then $\sum_{i=m}^{n} f(i) = 0$ and $\prod_{i=m}^{n} f(i) = 1$ (for any function f).

Given definitions

Definition (floor, ceiling). Let $x \in \mathbb{R}$. The **floor** of x is denoted $\lfloor x \rfloor$, and is defined as the greatest integer that is $\leq x$. The **ceiling** of x is denoted $\lceil x \rceil$, and is defined as the smallest integer that is $\geq x$.

Definition (factorial). Let $n \in \mathbb{N}$. The **factorial** of n is defined as $n! = \prod_{i=1}^{n} i$.

Definition (Big-Oh, Omega, Theta). Let $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}$.

$$g \in \mathcal{O}(f): \ \exists c, n_0 \in \mathbb{R}^+, \ \forall n \in \mathbb{N}, \ n \geq n_0 \Rightarrow g(n) \leq c \cdot f(n)$$

$$g \in \Omega(f): \ \exists c, n_0 \in \mathbb{R}^+, \ \forall n \in \mathbb{N}, \ n \geq n_0 \Rightarrow g(n) \geq c \cdot f(n)$$

$$g \in \Theta(f): \ g \in \mathcal{O}(f) \wedge g \in \Omega(f)$$

$$g \in \Theta(f): \ \exists c_1, c_2, n_0 \in \mathbb{R}^+, \ \forall n \in \mathbb{N}, \ n \geq n_0 \Rightarrow c_1 \cdot f(n) \leq g(n) \leq c_2 \cdot f(n) \qquad \text{(alternate definition)}$$

Warning: No work on this page will be graded.