UNIVERSITY OF TORONTO

Faculty of Arts and Science

term test #2, Version 1

CSC236F

Date: Thursday November 15, 6:10–7:00pm

Duration: 50 minutes

Instructor(s): Danny Heap

Examination Aids: pencils, pens, erasers, drinks, snacks

---

first and last names:

utorid:

student number:

Please read the following guidelines carefully!

- **Please write your name, utorid, and student number on the front of this exam.**
- This examination has **2** questions. There are a total of **5 pages, DOUBLE-SIDED**.
- Answer questions clearly and completely.
- You will receive 20% of the marks for any question you leave blank or indicate "I cannot answer this question."

Take a deep breath.
This is your chance to show us
How much you've learned.

We **WANT** to give you the credit

Good luck!

1. **[13 marks]** ($\approx$ 35 minutes)

Define $T(n)$ by:

$$T(n) = \begin{cases} 0 & \text{if } n < 2 \\ n + T(n-2) & \text{if } n \geq 2 \end{cases}$$

(a) **[3 marks]** Let $q \in \mathbb{N}$ and let $r \in \{0, 1\}$. Use the method of repeated substitution (unwinding) to find a conjecture for a closed form for $T(2q + r)$, that is some function c, using a fixed number of elementary operations, such that $c(2q+r) = T(2q+r)$. You may assume that if $n \in \mathbb{N}^+$, then $\sum_{i=1}^{i=n} i = n(n+1)/2$, if that assumption turns out to be useful.

sample solution: I notice that if $2q+r-2 = 2(q-1)+r$, and that $T(r) = 0$, so assuming $q \geq 1$ for the moment:

$$\begin{aligned}
T(2q+r) &= 2q + r + T(2(q-1)+r) && \text{\# definition of } T, q \geq 1 \\
&= 2q + r + 2(q-1) + r + T(2(q-2)+r) && \text{\# definition of } T, n \geq 2 \\
&= \\
&= \vdots \text{ (I see a pattern...)} \\
&= \\
&= \left(\sum_{i=1}^{i=q} 2i + r\right) + T(2(q-q)+r) = q(q+1) + qr && \text{\# using assumed sum of series} \\
&= q(q+r+1) = c(2q+r)
\end{aligned}$$

This form also agrees with $T(2q+r)$ when $q = 0$, so I'll conjecture $c(q) = q(q+r+1)$

- 2 marks for 2 successful substitutions
- 0.5 marks for successful derivation with summation
- 0.5 marks for replacing the summation with closed form

(b) **[6 marks]** Let $r \in \{0, 1\}$, and let function c be your conjecture for a closed form in part (a). Use induction on q to prove $\forall q \in \mathbb{N}, c(2q+r) = T(2q+r)$.

If you did not find a sucessful conjecture for c in part (a), for up to 4/6 marks you may show that $\forall q \in \mathbb{N}, T(2q+r) \geq q^2$. If you choose this option we will **not** grade any attempt to earn the full 6/6.

sample solution: (induction on q) I will prove this by induction on q. First, define $P(q) : c(2q+r) = T(2q+r)$.

base case: $T(2(0)+r) = 0$ by the recurrence definition, since $r < 2$. This agrees with $c(2(0)+r) = 0(0+1+r) = 0$, so $P(0)$ holds.

inductive step: Let $q \in \mathbb{N}$, and assume $P(q)$. I will show that $P(q+1)$ follows.

$$\begin{aligned}
T(2(q+1)+r) &= 2(q+1) + r + T(2q+r) && \text{\# since } 2(q+1) \geq 2 \\
&= 2(q+1) + r + c(2q+r) = 2(q+1) + r + q(q+1+r) && \text{\# by IH, } P(q) \\
&= (2+q)(q+1) + (q+1)r = (q+1)((q+1)+1+r) && \text{\# algebra...} \\
&= c(2(q+1)+r)
\end{aligned}$$

So $P(q+1)$ follows.

- 1 mark for defining, or consistently re-stating, predicate
- 1 mark for verifying base case
- 2 mark for introducing q and IH
- 1 mark for using the IH, and saying where you use it
- 1 mark for reaching conclusion

weaker claim $\forall q \in \mathbb{N}, T(2q+r) \geq q^2$: Define $P(q) : T(2q+r) \geq q^2$. Prove by simple induction on q that $\forall q \in \mathbb{N}, P(q)$.

base cases: By the function definition $T(2(0)+0) = T(2(0)+1) = 0 \geq 0^2$, so $P(0)$ holds.

inductive step: Let $q \in \mathbb{N}$. Assume $P(q)$. I will show that $P(q+1)$ follows.

$$\begin{aligned} T(2(q+1)+r) &= 2(q+1)+r+T(2q+r) && \text{\# since } q+1 \geq 1 \Rightarrow 2(q+1) \geq 2 \\ &\geq 2q+2+r+q^2 && \text{\# by IH} \\ &= q^2+2q+1+(1+r) \geq (q+1)^2 \quad \blacksquare && \text{\# since } 1+r \geq 0 \end{aligned}$$

So $P(q+1)$ follows.

- 1 mark for defining, or consistently re-stating, predicate
- 1 mark for verifying base case
- 1 mark for introducing q and IH
- 1 mark for using the IH and reaching conclusion

(c) **[4 marks]** Prove that $\forall n \in \mathbb{N}^+, T(n) - T(n-1) \geq 0$. In other words, prove that T is nondecreasing on $\mathbb{N}$. You may assume, as a consequence of the Quotient/Remainder Theorem, that $\forall n \in \mathbb{N}, \exists q, r \in \mathbb{N}, n = 2q+r \wedge 2 > r$.

sample solution: (without induction) Let $n \in \mathbb{N}^+$. Let $q, r \in \mathbb{N}, 2q+r = n \wedge 2 > r$. There are two cases to consider

case $r = 0$: Then $n = 2q+0$ and $n-1 = 2(q-1)+1$, so

$$\begin{aligned} T(n)-T(n-1) = T(2q+0)-T(2(q-1)+1) &= c(2q+1)-c(2(q-1)+0) && \text{\# by Q1(b)} \\ &= q(q+1+0)-(q-1)((q-1)+1+1) \\ &= q^2+q-q^2+1 = q+1 \geq 0 \end{aligned}$$

case $r = 1$: Then $n = 2q+1$ and $n-1 = 2q+0$, so

$$\begin{aligned} T(n)-T(n-1) = T(2q+1)-T(2q+0) &= c(2q+1)-c(2(q-1)+0) && \text{\# by Q1(b)} \\ &= q(q+1+1)-q(q+1+0) = q^2+2q-q^2-q = q \geq 0 \end{aligned}$$

In both possible cases $T(n) - T(n-1) \geq 0$ $\blacksquare$

- 1 mark for introducing n in terms of q r
- 1 mark for breaking into cases

- 1 mark for applying closed form
- 1 mark for getting inequality

alternative sample solution: (with induction) Define $P(n) : T(n) - T(n-1) \geq 0$. I will prove $\forall n \in \mathbb{N}^+, P(n)$ by complete induction.

inductive step: Let $n \in \mathbb{N}^+$. Assume $\bigwedge_{i=1}^{i=n-1} P(i)$. I will prove that $P(n)$ follows.

case $n \geq 3$: Then

$$\begin{aligned} T(n) - T(n-1) &= n + T(n-2) - (n-1) - T(n-3) && \text{\# by definition, since } n-1 \geq 2 \\ &\geq n - (n-1) = 1 \geq 0 && \text{\# by IH, since } 1 \leq n-2 < n \end{aligned}$$

So $P(n)$ follows in this case.

case $1 \leq n < 3$ From the definition $T(1) - T(0) = 0 \geq 0$ and $T(2) - T(1) = 2 \geq 0$, so $P(1)$ and $P(2)$ hold.

- 1 mark for predicate
- 1 mark for two base cases
- 1 mark for assumptions
- 1 mark for inductive step

2. **[5 marks]** (≈ 15 minutes)

Read over function **reversi** below:

```
1  def reversi(a_list: list) -> list:
2      """
3      Return a copy of a_list reversed.
4
5      Precondition: a_list is a Python list.
6
7      Postcondition: Returns a_list[ : : -1], which is a_list[ : ]
8                     in reverse order
9      """
10     if len(a_list) < 1:
11         return a_list[ : ]
12     else:
13         return reversi(a_list[1 : ]) + [a_list[0]]
```

Use induction on the size of **a_list** to prove that **reversi's** precondition, plus execution, implies its postcondition. Assume that the reverse-stride slice [1, 2, 3, 4][: : -1] returns [4, 3, 2, 1], that is the reverse of the original list.

sample solution: Define $P(n)$: For any list a_list with len(a_list) $= n$, reversi(a_list) returns a_list[: : -1]. I will prove $\forall n \in \mathbb{N}, P(n)$ by simple induction on n.

base case: By lines 9&10 reversi([ ]) returns [ ] = [ ][: : -1], so $P(0)$ holds.

inductive step: Let $n \in \mathbb{N}$. Assume $P(n)$. Let a_list be a list with len(a_list)$= n + 1$. Then by lines 11&12 reversi(a_list) returns:

`reversi(a_list[1 : ]) + [a_list[0]]`

which, by the inductive hypothesis, since len(a_list[1 :])$= n$, is:

`a_list[1: : -1] + [a_list[0]]`

... which is `a_list[ : : -1]`. So $P(n + 1)$ follows.

- 1 mark for defining, or repeatedly re-stating, predicate
- 2 marks for introducing n and the inductive hypothesis
- 1 mark for using the inductive hypothesis, and making it clear that they have used it
- 1 mark for reaching conclusion