COL100M Minor 2

Aman Godara

TOTAL POINTS

17 / 23

QUESTION 1

1 1 **1 / 1**

✓ + **1 pts** **Correct expression without syntax errors**

+ **0.5 pts** partially correct and/ or it has syntax errors

+ **0 pts** Incorrect or Not Attempted

QUESTION 2

2 2 **0 / 5**

+ **2 pts** Defined correct function (or code) for converting nat type to int type

+ **2 pts** Defined correct function (or code) for converting int type to nat type

+ **1 pts** Defined addnat correctly with help of conversion functions

+ **5 pts** Defined addnat correctly and completely

+ **1.5 pts** Base case is defined correctly (in a solution without type conversion)

+ **3 pts** Recursive case is defined (in a solution without type conversion)

+ **0.5 pts** Solution is complete and correct with part 5 (base case) and part 6 (recursive case)

✓ + **0 pts** **Incorrect or not Attempted**

+ **4 pts** Correct solution with missing type conversion

+ **2 pts** partially correct

+ **1 pts** partially correct

QUESTION 3

3 3 **2 / 2**

+ **2 pts** Correct

✓ + **0.5 pts** **Base case is correct**

✓ + **1 pts** **Hypothesis is correct**

+ **0.5 pts** Inductive step is correct

✓ + **0.5 pts** **Mention ,for k>0 P(k) holds in hypothesis step**

+ **0 pts** Didn't use Induction hypothesis in Induction step

+ **0 pts** Wrong/Blank

QUESTION 4

4 4 **2 / 2**

✓ + **2 pts** **Correct**

+ **1 pts** Definition is correct

+ **1 pts** Role stated is correct

+ **0.5 pts** The role stated isn't quite correct/complete.

+ **0.5 pts** Also mention, n> n0, c1>0 and c2>0

+ **0 pts** Role is not appropriate

+ **0 pts** Blank/Wrong

+ **0 pts** Click here to replace this description.

QUESTION 5

5 5 **3 / 3**

+ **3 pts** Correct

+ **1 pts** Ambiguous/imperfect explanation

+ **0 pts** Incorrect / Unattempted

✓ + **0.5 pts** **Correct basis step**

✓ + **0.5 pts** **Induction hypothesis written correctly**

✓ + **2 pts** **Correct explanation**

- **1 pts** Writing incorrect code

- **0.5 pts** Incorrect basis step

+ **1 pts** Imprecise proof

- **0.5 pts** Improper

+ **0.5 pts** Correct application of induction hypothesis

QUESTION 6

6 6 **4 / 5**

✓ + **0.5 pts** **Basis Correct**

✓ + **1.5 pts** **Induction Hypothesis correct (Strong**

induction or Weak induction with correct proof)

+ **1 pts** Induction Hypothesis weak with vague/ incorrect proof.

+ **1 pts** Weak Induction, but proof assumes strong Induction

- **0.5 pts** Minor errors in Inductive hypothesis

✓ + **1.5 pts** **Case "b is odd" Correct**

✓ + **1.5 pts** **Case "b is even" Correct**

+ **0.5 pts** Everything incorrect, but some structure to mathematical induction present.

+ **0.5 pts** Corrects 2 * b/2 to either include parenthesis or b mod 2 = 0. This is a Bonus. No penalty for not spotting this error.

+ **0 pts** Incorrect/Unattempted

✓ - **1 pts** **Minor Errors / incomplete proofs in the odd/even cases.**

You interchanged the poofs for even and odd cases.

QUESTION 7

7 7 5 / 5

✓ + **2 pts** **Tree correctly drawn with base case**

+ **1.5 pts** Base case missing in tree

+ **0.5 pts** tree drawn with branching

+ **1 pts** No derivation; just (log n) stated

✓ + **3 pts** **Derivation correct**

+ **2 pts** derived (log n) or (logn - 1) not (log n +1)

+ **0 pts** incorrect/no decision tree and time complexity

gradescope

COL100 – Minor exam 2

March 24, 2018

NAME:

AMAN GODARA

ID:

2017 TT 10876

1 Instructions

1. Write your answers *only in the space provided.* We are using software which automatically detects the answer region. If you write in the margins or in the wrong space, *your answer will not be graded.*
2. This exam requires you to write mathematical proofs and derivations. As repeatedly stressed in class, please write *precise* and *complete* answers. Anything short of this will result in loss of marks.
3. No calculators, phones, notes, or other resources are allowed. This is a closed book exam.
4. Time allocated for the exam: *1 hr.*

2 Exam begins here

1. (**1 point**) Suppose we define the following *recursive* data type to represent *natural numbers.*

```
type nat =
  | Zero
  | Succ of nat
```

Give an expression to construct the number 4 (that is, `let four = ...`).

let four = succ (succ (succ (succ (zero))))

2. (**5 points**) Write a function `addnat: nat -> nat -> nat`, that takes in two natural numbers (as defined in the previous question) and returns their sum. You may define additional functions if required.

~~let~~ ~~[illegible]~~

3. (**2 points**) State the steps involved in proving a statement $P(n)$ using mathematical induction.

Basis step : $P(n_0)$ is true $n_0 \in$ Integers
(This is shown by directly substituting n_0 in $P(n)$)

Inductive step : Assuming that $P(k)$ is true (where $k > n_0$)
we will try to prove that $P(k+1)$ is also true.

when $P(k+1)$ is proved true we can say that $P(n)$ is a true statement for all $n \geq n_0$

4. (**2 points**) Precisely define $\Theta(g(n))$. What is the role of $\Theta(g(n))$ in the asymptotic analysis of an algorithm (state in just 1 or 2 sentences *only*).

$\Theta(g(n)) = \{ f(n)$ there exist $c_1 > 0$, $c_2 > 0$ s.t. $0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$ for $n \geq n_0 \}$

$\Theta(g(n))$ is used to estimate worst time and best time taken by an algorithm.
It measures upper bound and lower bound of the runtime of an algorithm.

5. **(3 marks)** Prove that the following function sum correctly returns the sum of 2 positive integers. If it does not return the sum. state and correct the mistake and then provide a correctness proof.

```
let rec sum a b =
  if a = 0 then b
     else (sum (a-1) b) + 1;;
```

P(n) states that sum n m = n + m
(where m is any arbitary constant)

basis step: P(1) is true as
sum 1 m
= m + 1, which is equal to 1+m

Inductive step: P(k) is assumed to be true
ie sum k m = k + m

Now P(k+1),
sum (k+1) m
= (sum (k+1-1) m) + 1
= k + m + 1
= (k+1) + m
= addition of (k+1) & m

⇒ P(k+1) is true, hence sum a b returns a+b (where b is any arbitary constant)

⇒ we have proved assuming second input as any arbitary constant

⇒ Using same procedure we can show that sum (n) (m) gives m+n for some arbitary constant n.

hence for every n, m (n, m > 0) sum n m gives n + m

6. (**5 marks**) Prove that the following function `mult` correctly returns the product of 2 positive integers. If it does not return the product, state and correct the mistake and then provide a correctness proof.

```
let rec mult a b =
  if b = 0 then 0
  else if b = 2 * b/2 then 2 * mult a (b/2)
  else a + mult a (b-1);;
```

P(n) states that mult a b = $a \times b$ for any arbitary constant a

basis step : P(0) is true as

mult a 0

= 0

= multiplication of a and 0 = 0

strong Inductive step : P(m) is true $\forall$ $(0 < m \leq k)$ —(i)

= mult a m

= $a \times m$

Now P(k+1),

mult a (k+1)

$= \begin{cases} (2) \text{ mult } a \left(\frac{k+1}{2}\right) & \text{if } k \text{ is even} \\ a + \text{mult } (a)\left(\frac{k-1}{}\right) & \text{if } k \text{ is odd} \end{cases}$

∴ (by strong induction) (using i)

$= \begin{cases} (2)\left(a \cdot \left(\frac{k+1}{2}\right)\right) & \text{if } k \text{ is even} \\ a + (a)\left(\frac{k+1-1}{}\right) & \text{if } k \text{ is odd} \end{cases}$

$\begin{cases} (a)(k+1) & \text{if } k \text{ is even} \\ a(k+1) & \text{if } k \text{ is odd} \end{cases}$

= multiplication of a and k+1

we have proved for any arbitary constant a hence P(n) is true for all $n \geq 0$, $n \in$ Integers

mult a b always gives $a \times b$.

7. **(5 marks)** Derive an expression in terms of the Θ notation for the *runtime complexity* of the following function power that computes a positive power of a positive integer. Use the recursion tree method. Show *at least* the first three levels of the tree (including the root).

```
let rec power a n =
   if n = 0 then 1
   else
     let half_pow = power a (n/2) in
     if n mod 2 == 0 then half_pow * half_pow
     else a * half_pow * half_pow;;
```

n (root) [ ] → represents greatest integer function

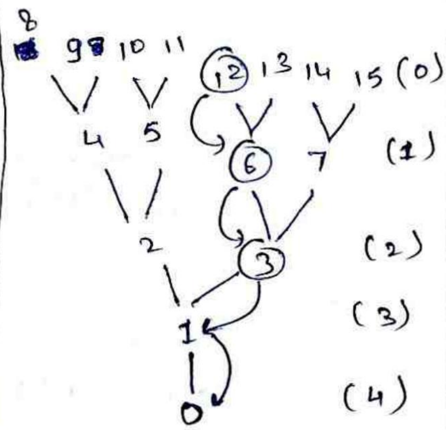
$\left[\frac{n}{2}\right]$ (level 1)

$\left[\frac{\left[\frac{n}{2}\right]}{2}\right]$ (level 2)

$\left[\frac{\left[\frac{\left[\frac{n}{2}\right]}{2}\right]}{2}\right]$ (level 3)

⋮

0



fig(1)

~~for eg:~~ if input is 12 fig(1) shows how ~~recursive~~ function will proceed

$T(n) = \left\{ p+1, \text{ where } 2^p \le n < 2^{p+1} \;\; \forall\, n > 1 \right\}$ (by observation)

$2^p \le n < 2^{p+1}$ (take logarithm)

$p \log 2 \le \log n < (p+1)\log 2$

$p+1 \le \frac{\log n}{\log 2} + 1$

$p+1 > \frac{\log n}{\log 2}$

c_1

$\frac{\log n}{\log 2} < \underbrace{\cancel{p+1}}_{T(n)} \le \frac{\log n}{\log 2} + 1$

$n_0 = 2$

$T(n) = \Theta(\underbrace{\log n}_{g(n)})$

$\frac{\log n}{\log 2} + 1 < c_2 \frac{\log n^2}{\log 2}$

$\frac{\log n}{\log 2} + 1 < \frac{(2)}{\log(2)}(\log n)$

$\forall\, n > 0$