COL202 Quiz 1

Aaveg Jain

TOTAL POINTS

0 / 5

QUESTION 1

1 Loop invariant 0 / 5

✓ **+ 0 pts** *Incorrect/Not attempted*

+ 0.5 pts The algorithm returns a linked list of length $$2^{(length(l))}$$.

+ 1 pts Invariant- for all $$0<=i<=len(l)$$: $$len(l')=2^i$$ at the end of $$i^{th}$$ iteration.

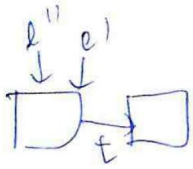
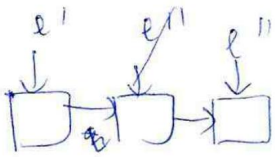
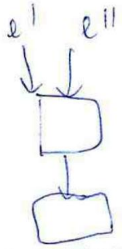
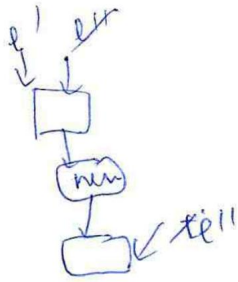
+ 0.5 pts Proof by induction on the outer loop.

+ 1 pts Base Case

+ 0.5 pts Induction Hypothesis

+ 1 pts Induction Step

+ 0.5 pts Conclusion- At the end of the algorithm the value of $$i = length(l)$$ and hence the length of $$l'$$ is $$2^{length(l)}$$.

COL202: Discrete Mathematical Sturctures. I semester, 2022-23.
Quiz 1, 24 August 2022, Maximum Marks: 5

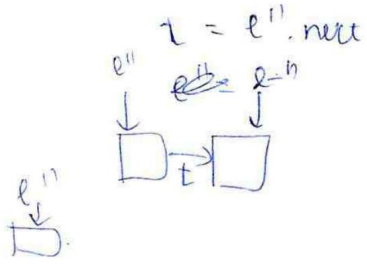
Name Aarg Jain	**Ent. No.** 2021CS10073

Important: Answer within the box. Anything written outside the box will be treated as rough work.

Problem 1 (5 marks)

What does the algorithm on the right return? Prove that your answer is correct by defining an appropriate loop invariant and proving its correctness by induction. Assume that the last node of a linked list points to NULL.

Require: : Given a linked list ℓ.
1: initialise a list ℓ' containing 1 node
2: **while** ℓ is not NULL **do**
3: $\quad \ell'' \leftarrow \ell'$
4: $\quad$ **while** ℓ'' is not NULL **do**
5: $\quad\quad t \leftarrow \ell''$.next
6: $\quad\quad$ Insert a new node after ℓ''
7: $\quad\quad \ell'' \leftarrow t$
8: $\quad$ **end while**
9: $\quad \ell \leftarrow \ell$.next
10: **end while**
11: Return ℓ'



the inner loop adds a new ~~element~~ node to list ℓ' which is next to ℓ first element. This loop is performed as many times as length of ℓ. Hence n nodes will be added to ℓ' ($=n$)

We use proof by induct

~~P(i): at the~~

Base case: at the end of 1st iteration, one new ℓ' is added to ℓ'.

~~Ind~~ IH: P(i) at end of i^{th} iteration, i new nodes added to ℓ' at beginning.