## **بسمه تعالی**



**گزارش‌ تمرین4 سوالات تحلیلی**

**ساختمان داده**

محسن کربلائی امینی، 98242128

آذر ۱۴۰2

## **سوال 1:**

**struct node {**

**int data;**

**node\* next;**

**}**

**node\* mianeLinkedList (linkedlist ll) {**

**node\* result=ll.L;**

**node\* nextCandidate=ll.L->next;**

**counter=0;**

**for (node\* i = ll.L ; i !=null ; i=i->next, counter=counter++%2) {**

**if (counter == 1) {**

**result=nextCandidate;**

**nextCandidate=ll[result]->next;**

**}**

**}**

**return result;**

**}**

## **سوال 2:**

**Node\* reverse(Node\* node)**

**{**

**if (node == NULL)**

**return NULL;**

**if (node->next == NULL) {**

**head = node;**

**return node;**

**}**

**Node\* node1 = reverse(node->next);**

**node1->next = node;**

**node->next = NULL;**

**return node;**

**}**

## **سوال 3:**

**node\* merge(node\* nHead, node\* mHead ) {**

**node\* previousNode=nHead; *// just an assumption that nhead's value is lower.***

**node\* result=nHead;**

**for (int i=0 ; i < m+n ; i++) {**

***/\* compare paires of n and m then acquire***

***the lowest as a node to the new list***

***and go on.***

***\*/***

**if (nHead->value < mHead->value) {**

**if (nHead != previousNode)**

**previousNode->next=nHead;**

**nHead=nHead->next;**

**} else {**

**if (i==0)**

**result=mHead;**

**previousNode->next=mHead;**

**mHead=mHead->next;**

**}**

**}**

**return result;**

**}**

## **سوال 4:**

**node\* sortAbsoluteSortedLinkedList (node\* head) {**

**node\* result=head;**

**node\* previousNode=head;**

**for (node\* L=head ; L != NULL ; ) {**

***// the iterations are done in if and else clauses***

**if (L->value < 0){**

***// iterate the list from first node up to end***

***// put every negative node to the first of the list***

**previousNode->next=L->next;**

**node\* temp = L->next;**

**L->next = result;**

**result=L;**

**L=temp;**

**}else {**

**previousNode=previousNode->next;**

**L=L->next;**

**}**

**}**

**return result;**

**}**

## **سوال 5:**

**bool hasCircle(node\* head) {**

**for (node\* L=head ; L != NULL ; L=L->next) {**

**if (L->next == head->next)**

**{**

**L->next = null;**

**return true;**

**}**

**}**

**return false;**

**}**