struct ListNode\* reverseBetween(struct ListNode\* head, int left, int right) {

    if(left==1 && right==1)

    {

        return(head);

    }

    struct ListNode\* Previous\_Node=NULL;

    struct ListNode\* Current\_Node=head;

    struct ListNode\* Next\_Node=head;

    struct ListNode\* newHead=head;

    struct ListNode\* newNull;

    int position;

    if(left!=-1)

    {

        for(position=1; position<left; position++)

        {

            if(position==left-1)

            {

                newHead=Current\_Node;

            }

            Current\_Node=(\*Current\_Node).next;

        }

    }

    Previous\_Node=Current\_Node;

    newNull=Current\_Node;

    Current\_Node=(\*Current\_Node).next;

    for(position=left+1; position<=right; position++)

    {

        Next\_Node=(\*Current\_Node).next;

        (\*Current\_Node).next=Previous\_Node;

        Previous\_Node=Current\_Node;

        if(position==right)

        {

            if(left==1)

            {

                head=Current\_Node;

            }

            else

                (\*newHead).next=Current\_Node;

        }

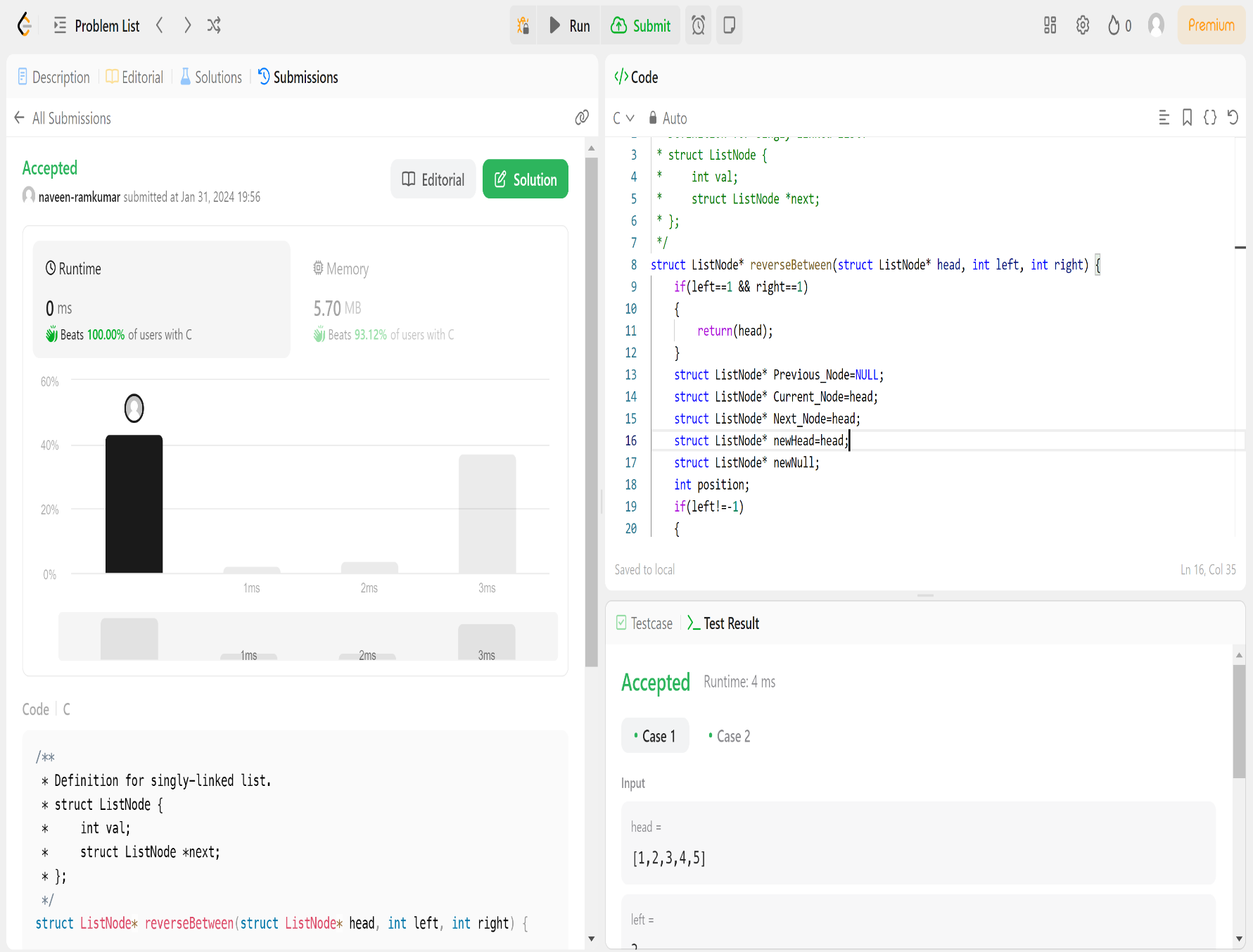
        Current\_Node=Next\_Node;

    }

    (\*newNull).next=Current\_Node;

    return(head);

}

****