**Blockchain Technology Lab**

**Lab – 1**

**Aim : Implement a doubly linked list in append mode only.**

**Code :**

*#include* <bits/stdc++.h>

using *namespace* std;

*class* Student

{

*public:*

*int* adm\_no;

*int* marks;

    string branch;

    Student(*int* a, *int* m, string b)

    {

*this*->adm\_no = a;

*this*->marks = m;

*this*->branch = b;

    }

};

*class* node

{

*public:*

    Student data;

    node \*next;

    node \*prev;

    node(Student val) : data(val)

    {

*this*->next = NULL;

*this*->prev = NULL;

    }

};

*void* insertAtTail(node *\*&*head, Student val)

{

    node \*new\_node = *new* node(val);

*if* (head == NULL)

    {

        head = new\_node;

*return*;

    }

    node \*temp = head;

*while* (temp->next != NULL)

    {

        temp = temp->next;

    }

    temp->next = new\_node;

    new\_node->prev = temp;

}

*void* display(node *\**head)

{

    node \*temp = head;

*while* (temp != NULL)

    {

        cout << "Adm No: " << temp->data.adm\_no << ", Marks: " << temp->data.marks << ", Branch: " << temp->data.branch << "\n";

        temp = temp->next;

    }

}

*int* main()

{

    node \*head = NULL;

*int* choice;

*while* (true)

    {

        cout << "\nMenu:\n";

        cout << "1. Insert\n";

        cout << "2. View\n";

        cout << "3. Exit\n";

        cout << "Enter your choice: ";

        cin >> choice;

*switch* (choice)

        {

*case* 1:

        {

*int* adm\_no, marks;

            string branch;

            cout << "Enter admission number: ";

            cin >> adm\_no;

            cout << "Enter marks: ";

            cin >> marks;

            cout << "Enter branch: ";

            cin >> branch;

            insertAtTail(head, Student(adm\_no, marks, branch));

*break*;

        }

*case* 2:

        {

            display(head);

*break*;

        }

*case* 3:

        {

*return* 0;

        }

*default*:

            cout << "invalid option" << endl;

*break*;

        }

    }

*return* 0;

}

**Output :**

**A computer screen shot of a black screen

Description automatically generated**