```
#include<iostream>
using namespace std;

struct SLLNode* createSLL(int cnt, struct SLLNode *head);
void displaySLL(struct SLLNode *head);

void A_U_B();
void A_int_B();
void A_Min_B();
void B_Min_A();
void U_Min_A_U_B();



struct SLLNode
{
  char data;
  struct SLLNode *next;

}*headU, *headA, *headB;

int main()
{
  int i,no;

  cout<<"\n\n\t How many Linked Lists: ";
  cin>>no;

  headU = headA = headB = NULL;

  for(i=1; i<=no; i++)
  {
```

```
  if(i == 1)
  {
    cout<<"\n\n\t Enter number of Students : ";
    headU = createSLL(10, headU);
      cout<<"\n";
    displaySLL(headU);
  }

  if(i == 2)
  {
    cout<<"\n\n\t Enter Students who like Vanilla Icecreme: ";
    headA = createSLL(3, headA);
      cout<<"\n";
    displaySLL(headA);
  }

  if(i == 3)
  {
    cout<<"\n\n\t Enter Students who like Butterscotch Icecreme: ";
    headB = createSLL(3, headB);
      cout<<"\n";
    displaySLL(headB);
  }
}

        cout<<"\n\n Input Sets:-------------------------";

        cout<<"\n\n Set 'U': ";
        displaySLL(headU);

        cout<<"\n\n Set 'A': ";
```

```
        displaySLL(headA);

        cout<<"\n\n Set 'B': ";
        displaySLL(headB);

        cout<<"\n\n Output Sets:----------";
        A_U_B();
        A_int_B();
        A_Min_B();
        B_Min_A();
        U_Min_A_U_B();


  cout<<"\n\n";
  return 0;
}

//.....................................................Function to create Linked List as Sets.


struct SLLNode* createSLL(int cnt, struct SLLNode *head)
{
  int i;
  struct SLLNode *p, *newNode;

  for(i=0; i<cnt; i++)
  {
     newNode = new(struct SLLNode);              //  1. DMA

     cout<<"\n\t Enter Student Initial: ";  //  2. Data & Address Assignment
     cin>>newNode->data;
```

```
      newNode->next = NULL;

      if(head == NULL)                                  //  3. Add node in the list
      {
        head = newNode;
           p = head;
      }
      else
      {
        p->next = newNode;
             p = p->next;
      }
   }

   return head;
}




//.....................................................Function to display Linked Lists as Sets.

void displaySLL(struct SLLNode *head)
{
   struct SLLNode *p;

   p = head;
   while(p != NULL)
   {
     cout<<"  "<<p->data;
     p = p->next;
   }
```

```
}



//.....................................................Function for Set A U B .



void A_U_B()
{
  int i,j;
  char a[10];
  struct SLLNode *p, *q;

  i = 0;   //Index of Resultant Array
  p = headA;      //pointer to Set 'A'
  q = headB;   //pointer to Set 'B'

  while(p != NULL && q != NULL)
  {
    if(p->data == q->data)
    {
      a[i] = p->data;
      i++;
        p = p->next;
        q = q->next;
    }
    else
    {
      a[i] = p->data;
      i++;
        p = p->next;
```

```
    }
  }
  if(p == NULL)   //Set 'A' copied completely
  {
    while(q != NULL)       //Copy remaining elements of Set 'B'
    {
      a[i] = q->data;
      i++;
        q = q->next;
    }
  }

  if(q == NULL)   //Set 'B' copied completely
  {
    while(p != NULL)       //Copy remaining elements of Set 'A'
    {
      a[i] = p->data;
      i++;
        p = p->next;
    }
  }


  cout<<"\n\n\t Set A U B: ";
  for(j=0; j < i; j++)
    cout<<" "<<a[j];

}


//.....................................................Function for Set A ^ B .
```

```
void A_int_B()
{
  int i,j;
  char a[10];
  struct SLLNode *p, *q;

  i = 0;   //Index of Resultant Array
  p = headA;      //pointer to Set 'A'

  while(p != NULL)
  {
    q = headB;   //pointer to Set 'B'
    while(q != NULL)
    {
      if(p->data == q->data)
      {
        a[i] = p->data;
        i++;
      }
      q = q->next;
    }
    p = p->next;
  }

  cout<<"\n\n\t Set A ^ B: ";
  for(j=0;j < i;j++)
    cout<<" "<<a[j];

}
```

```
//.....................................................Function for Set A - B .

void A_Min_B()
{
  int i,j,flag;
  char a[10];
  struct SLLNode *p, *q;

  i = 0;   //Index of Resultant Array
  p = headA;      //pointer to Set 'A'

  while(p != NULL)
  {
    flag = 0;
    q = headB;   //pointer to Set 'B'

    while(q != NULL)
    {
      if(p->data == q->data)
      {
        flag = 1;
      }
      q = q->next;
    }
    if(flag == 0)
    {
      a[i] = p->data;
        i++;
    }
    p = p->next;
```

```
  }

  cout<<"\n\n\t Set A - B: ";
  for(j=0; j < i; j++)
    cout<<" "<<a[j];
}



//.....................................................Function for Set B - A.

void B_Min_A()
{
  int i,j,flag;
  char a[10];
  struct SLLNode *p, *q;

  i = 0;   //Index of Resultant Array
  q = headB;      //pointer to Set 'B'

  while(q != NULL)
  {
    flag = 0;
    p = headA;   //pointer to Set 'A'
    while(p != NULL)
    {
      if(q->data == p->data)
      {
        flag = 1;
      }
      p = p->next;
```

```
    }
    if(flag == 0)
    {
      a[i] = q->data;
        i++;
    }
    q = q->next;
  }

  cout<<"\n\n\t Set B - A: ";
  for(j=0; j < i; j++)
    cout<<" "<<a[j];
}


//.....................................................Function for Set U - (A U B).

void U_Min_A_U_B()
{
   int i,j,flag;
  char a[10];
  struct SLLNode *p, *q, *r;

  i = 0;   //Index of Resultant Array
  p = headU;      //pointer to Set 'U'

  while(p != NULL)
  {
    flag = 0;
    q = headA;   //pointer to Set 'A'
    r = headB;   //pointer to Set 'B'
```

```
    while(q != NULL)
    {
      if(p->data == q->data)
      {
        flag = 1;
      }
      q = q->next;
    }
    while(r != NULL)
    {
      if(p->data == r->data)
      {
        flag = 1;
      }
      r = r->next;
    }
    if(flag == 0)
    {
      a[i] = p->data;
        i++;
    }
    p = p->next;
  }

  cout<<"\n\n\t Set U - (A U B): ";
  for(j=0; j < i; j++)
    cout<<" "<<a[j];
}
```