# MATRIX MULTIPLICATION AND LINKEDLIST

**1)#include <stdio.h>**

**#include<stdlib.h>**

**int main() {**

**int a[3][3];**

**int b[3][3];**

**int mul[3][3];**

**int \*p;**

**printf("\nEnter the 3x3 matrix A:-\n");**

**for(int i = 0; i < 3; i++)**

**{**

**for (int j = 0; j < 3; j++)**

**{**

**scanf("%d",&a[i][j]);**

**}**

**}**

**printf("\nEnter the 3x3 matrix B:-\n");**

**for(int i = 0; i < 3; i++)**

**{**

**for (int j = 0; j < 3; j++)**

**{**

**scanf("%d",&b[i][j]);**

**}**

**}**

**p = (int\*)malloc(9\*sizeof(int));**

**p =&mul[0][0];**

**for(int i = 0; i < 3; i++)**

**{**

**for(int j = 0; j < 3 ;j++)**

**{**

**mul[i][j]=0;**

**for(int k = 0; k < 3;k++)**

**{**

**mul[i][j]+=a[i][k]\*b[k][j];**

**}**

**}**

**}**

**printf("\nThe multiplication of the matrix a and b is:-\n");**

**for(int i = 0; i < 3; i++)**

**{**

**for (int j = 0; j < 3; j++)**

**{**

**printf("%d\t",\*(p+i\*3+j));**

**}**

**printf("\n");**

**}**

**return 0;**

**}**

2. You are given a task with creating a simple student management system using arrays that will allow the user to manage student names. Implement the following operations on a list of student names using switch-case and arrays. After every operation, display the current list of students. The operations to implement are: (i) Creation of the list: Allow the user to create a list of student names by entering them one by one. (ii) (iii) (iv) (v) Insertion of a new student: Insert a new student's name into a specific position in the list. The user should provide the name and the index at which it should be inserted. Deletion of a student: Delete a student's name from the list based on their position or name. Ask the user whether they want to delete by name or by index. Traversal of the list: Display all the student names in the current order. Search for a student: Search for a student's name in the list and display whether or not the student is found, along with their position if present.

#include <stdio.h>

#include<stdlib.h>

typedef struct node

{

char data[20];

struct node \*next;

}node;

int main()

{

node \*a,\*b,\*c,\*head;

node \*temp;

int op;

char name[20];

a = (node\*)malloc(sizeof(node));

b = (node\*)malloc(sizeof(node));

c = (node\*)malloc(sizeof(node));

printf("\nEnter the name of the studdents (MAX 3):-\n");

scanf("%s",a->data);

scanf("%s",b->data);

scanf("%s",c->data);

head = a;

a->next = b;

b->next = c;

c->next = NULL;

printf("\nOperations you can make in the list:-\n a)For Insertion enter 1\n b)For deletion enter 2\n c)For Traversal enter 3\n d)For searching enter 4\n");

scanf("%d",&op);

switch(op)

{

case 1:

printf("\nYou've chose insertion\n");

node \*d;

d = (node\*)malloc(sizeof(node));

printf("\nEnter the name you want to insert:-");

scanf("%s",d->data);

d->next = head;

head = d;

printf("\nThe new linked list after insertion is\n");

temp = head;

while(temp != NULL)

{

printf("%s\t",temp->data);

temp = temp->next;

}

break;

case 2:

printf("\nYou've choesen deletion\n");

printf("Enter the name you want to delete\n");

scanf("%s",name);

if(name == a->data)

{

head = a->next;

free(a);

printf("\nThe new linked list after deletion is\n");

temp = head;

while(temp != NULL)

{

printf("%s\t",temp->data);

temp = temp->next;

}

}

else if(name == b->data)

{

head->next = c;

head->next->next=NULL;

free(b);

printf("\nThe new linked list after deletion is\n");

temp = head;

while(temp != NULL)

{

printf("%s\t",temp->data);

temp = temp->next;

}

}

else(name == c->data);

{

head->next->next = NULL;

free(c);

printf("\nThe new linked list after deletion is\n");

temp = head;

while(temp != NULL)

{

printf("%s\t",temp->data);

temp = temp->next;

}

}

break;

case 3:

printf("\nTraversal of the linked list is\n");

temp = head;

while(temp != NULL)

{

printf("%s\t",temp->data);

temp = temp->next;

}

break;

case 4:

printf("\nEnter the name you want to search:-");

scanf("%s\n",name);

if(name == a->data)

{

printf("%s",a->data);

}

else if(name == b->data)

{

printf("%s",b->data);

}

else if(name == c->data)

{

printf("%s",c->data);

}

else

{

printf("Name is not there");

}

break;

default:

printf("\nYou chose nothing");

}

return 0;

}