AKSHAY RAJ SINGH

BT22CSH054

CODE:-

1. #include<iostream>

using namespace std;

class element{

public:

int i;

int j;

int x;

};

class sparse{

public:

int m;

int n;

int num;

element \*e;

sparse(int m,int n,int num){

this->m=m;

this->n=n;

this->num=num;

this->e=new element[this->num];

}

};

void create(sparse\* s){

cout<<"enter the non-zero element:";

for(int a=0;a<s->num;a++){

cin>>s->e[a].i>>s->e[a].j>>s->e[a].x;

}

}

void display(sparse \*s){

int k=0;

for(int a=0;a<s->m;a++){

for(int b=0;b<s->n;b++){

if(a==s->e[k].i&&b==s->e[k].j)

cout<<s->e[k++].x<<" ";

else

cout<<"0 ";

} cout<<endl;

}

}

int main(){

int m,n,num;

cout<<"enter the dimension of matrix:";

cin>>m>>n;

cout<<"enter the no. of non zero element:"<<endl;

cin>>num;

sparse \*s=new sparse(m,n,num);

create(s);

display(s);

return 0;}

2. #include <stdio.h>

#include <stdlib.h>

struct Node {

int digit;

struct Node\* next;

};

struct Node\* createNode(int digit) {

struct Node\* newNode = (struct Node\*)malloc(sizeof(struct Node));

newNode->digit = digit;

newNode->next = NULL;

return newNode;

}

void insertDigit(struct Node\*\* head, int digit) {

struct Node\* newNode = createNode(digit);

newNode->next = \*head;

\*head = newNode;

}

struct Node\* addNumbers(struct Node\* num1, struct Node\* num2) {

struct Node\* result = NULL;

int carry = 0;

while (num1 || num2 || carry) {

int sum = carry;

if (num1) {

sum += num1->digit;

num1 = num1->next;

}

if (num2) {

sum += num2->digit;

num2 = num2->next;

}

carry = sum / 10;

sum = sum % 10;

insertDigit(&result, sum);

}

return result;

}

void displayNumber(struct Node\* head) {

struct Node\* current = head;

while (current != NULL) {

printf("%d", current->digit);

current = current->next;

}

printf("\n");

}

int main() {

struct Node\* num1 = NULL;

struct Node\* num2 = NULL;

char str1[1000];

printf("Enter the first long integer: ");

scanf("%s", str1);

for (int i = 0; str1[i] != '\0'; ++i) {

int digit = str1[i] - '0';

insertDigit(&num1, digit);

}

char str2[1000];

printf("Enter the second long integer: ");

scanf("%s", str2);

for (int i = 0; str2[i] != '\0'; ++i) {

int digit = str2[i] - '0';

insertDigit(&num2, digit);

}

struct Node\* sum = addNumbers(num1, num2);

printf("Sum: ");

displayNumber(sum);

return 0;

}