Set1 :

1. Print the word with odd letters as

P M

R A

O R

G

O R

R A

P M

#include<iostream.h>

#include<conio.h>

void main(){

clrscr();

int n;

cout<<"\n enter size:";

cin>>n;

char \*a=new char [n];

cout<<"\n Enter name:";

cin>>a;

for(int i=0;i<n;i++)

{

for(int j=0;j<n;j++)

{

if((i==j)||(i+j==n-1))

cout<<a[j];

cout<<"\t";

}

cout<<"\n";

}

getch();

}

2. Given a set of numbers like <10, 36, 54,89,12> we want to find sum of weights based on the following conditions  
    1. 5 if a perfect square  
    2. 4 if multiple of 4 and divisible by 6  
    3. 3 if even number

And sort the numbers based on the weight and print it as follows

<10,its\_weight>,<36,its weight><89,its weight>

Should display the numbers based on increasing order.

#include<stdio.h>

#include<math.h>

int getWeight(int n)

{

int w=0;

float root=sqrt(n);

if(root==ceil(root))

    w+=5;

if(n%4==0&&n%6==0)

    w+=4;

if(n%2==0)

    w+=3;

return w;

}

void main()

{

int nums[15];

int ws[15];

int i,j,t,n;

printf("Enter number of numbers");

scanf("%d",&n);

printf("\nEnter numbers");

for(i=0;i<n;i++)

    scanf("%d",&nums[i]);

for(i=0;i<n;i++)

    ws[i]=getWeight(nums[i]);

printf("\nBefore sorting:\n");

for(i=0;i<n;i++)

    printf("%d:%d\t",nums[i],ws[i]);

for(i=0;i<n;i++)

    for(j=0;j<n-i-1;j++)

        if(ws[j]>ws[j+1])

            {

            t=ws[j+1];

            ws[j+1]=ws[j];

            ws[j]=t;

            t=nums[j+1];

            nums[j+1]=nums[j];

            nums[j]=t;

            }

printf("\nSorted:\n");

for(i=0;i<n;i++)

    printf("%d:%d\t",nums[i],ws[i]);

}

3. Save the string “WELCOMETOZOHOCORPORATION” in a two dimensional array and search for substring like “too” in the two dimensional string both from left to right and from top to bottom.

weLCO

METOZ

OHOCO

RPORA

TIOn

And print the start and ending index as

Start index : <1,2>

End index: <3, 2>

#include <stdio.h>

**main**()

{

**char** \*c="welcometozohocorporation";

**char** array[5][5];

**int** x;

**int** y;

**int** i=0;

**for**(x=0;x<5;x++)

{

**for**(y=0;y<5;y++)

{ array[x][y]=c[i]; //intialising the characters of string to the 2d array

i++;

}}

**for**(x=0;x<5;x++)

{

**for**(y=0;y<5;y++)

{

printf("array[%d][%d]=%c\n",x,y,array[x][y]);//printing the 2d array to make sure that it has been initialised with the string characters

}

}

**for**(x=0;x<5;x++)

{

**for**(y=0;y<5;y++)

{

//check for left to right existence

**if**((array[x][y]=='t')&&(array[x][y+1]=='o')&&(array[x][y+2]=='o'))

{

printf("it starts at %d,%d",x,y);

printf("\n");

printf("it ends at %d,%d",x,y+2);

}

printf("\n");

//check for top to bottom existence

**if**((array[x][y]=='t')&&(array[x+1][y]=='o')&&(array[x+2][y]=='o'))

{

printf("it starts at %d,%d",x,y);

printf("\n");

printf("it ends at %d,%d",x+2,y);

}

}}

}

5. Given a two dimensional array of string like

<”luke”, “shaw”>

<”wayne”, “rooney”>

<”rooney”, “ronaldo”>

<”shaw”, “rooney”>

Where the first string is “child”, second string is “Father”. And given “ronaldo” we have to find his no of grandchildren Here “ronaldo” has 2 grandchildren. So our output should be 2.

#include <stdio.h>

#include <string.h>

int main()

{

char a[10][100],b[10][100],str[100];

int num,k;

scanf("%d",&num);

scanf("%s",&str);

for(int i=0;i<num;i++)

{

scanf("%s%s",&a[i],&b[i]);

if(strcmp(str,b[i]) == 0)

k = i;

}

int count = 0;

for(int i=0;i<num;i++)

{

if(strcmp(a[k],b[i]) == 0)

count++;

}

printf("%d",count);

return 0;

}

6.Finding the sec largest negative element in an array:

#include<stdio.h>

#include<conio.h>

void main()

{

int \*a,min,i,j,n,temp=0,b,\*c;

clrscr();

printf("emtr the size of the array:");

scanf("%d",&n);

a=(int \*)malloc(sizeof(int) \* n);

c=(int \*)malloc(sizeof(int) \* n);

printf("enter the elements:");

for(i=0;i<n;i++)

{

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

}

for(i=0;i<n;i++)

{

c[i]=a[i];

}

printf(" original c array:");

for(i=0;i<n;i++)

{

printf("%d",c[i]);

}

min=a[0];

for(i=0;i<n;i++)

{

for(j=i+1;j<n;j++)

{

if(a[i]>a[j])

{

b=a[i];

a[i]=a[j];

a[j]=b;

}

}

}

printf("sorted c array is:");

for(i=0;i<n;i++)

{

printf("%d",a[i]);

}

for(i=0;i<n;i++)

{

if(min>a[i])

{

min=a[i];

j=i;

}

}

printf("min :%d\n",min);

for(i=0;i<n;i++)

{

if(a[i]<0 && a[i]>min && j!=i)

{

temp=a[i];

}

if(min==a[i])

{

temp=a[i];

}

}

printf("\ntemp:%d",temp);

for(i=0;i<n;i++)

{

if(temp==c[i])

{

printf("temp at position at :%d",i);

}

}

getch();

}

Set 2:

1) Alternate sorting: Given an array of integers, rearrange the array in such a way that the first element is first maximum and second element is first minimum.

Eg.) Input : {1, 2, 3, 4, 5, 6, 7}

Output : {7, 1, 6, 2, 5, 3, 4}

#include<stdio.h>

#include<conio.h>

void main()

{

int \*a,n,i,j,temp;

clrscr();

printf("enter the size of the array:");

scanf("%d",&n);

a=(int \*)malloc(sizeof(int) \*n);

printf("enter the elements:");

for(i=0;i<n;i++)

{

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

}

for(i=0;i<n;i++)

{

for(j=0;j<i;j++)

{

if(a[i]<a[j])

{

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

}

printf("sorted array:");

for(i=0;i<n;i++)

{

printf("%d",a[i]);

}

printf("array:");

for(i=0,j=n-1;i<n,j>=0;i++,j--)

{

if(i<j)

printf("%d%d",a[j],a[i]);

if(i==j)

printf("%d",a[i]);

}

getch();

}

3) Form a number system with only 3 and 4. Find the nth number of the number system.  
Eg.) The numbers are: 3, 4, 33, 34, 43, 44, 333, 334, 343, 344, 433, 434, 443, 444, 3333, 3334, 3343, 3344, 3433, 3434, 3443, 3444 ….

#include<iostream.h>

#include<conio.h>

void num(int n)

{

int number[100],i=0;

n=n+1;

while(n!=1)

{

if(n%2==0)

number[i++]=3;

else

number[i++]=4;

n/=2;

}

while(i--)

cout<<number[i];

}

int main()

{

int n;

cin>>n;

num(n);

getch();

return 200;

}

**Output**

5

43

Set 3:

1. Write a program to give the following output for the given input

Eg 1: Input: a1b10

Output: abbbbbbbbbb

Eg: 2: Input: b3c6d15

Output: bbbccccccddddddddddddddd

The number varies from 1 to 99.

#include<stdio.h>

#include<string.h>

int main()

{

char a[100],t;

int i,j,x,f;

printf("Enter the string:");

scanf("%s",&a);

for(i=0;i<strlen(a);i=i+2)

{

t=a[i];

if((a[i+2]-48)>=0 && (a[i+2]-48)<=9)

{

int x;

x=((a[i+1]-48)\*10)+(a[i+2]-48);

for(f=0;f<x;f++)

{

printf("%c",t);

}

i++;

}

else

{

for(j=0;j<a[i+1]-48;j++)

{

printf("%c",t);

}

}

}

return 0;

}

**Reverse process**

**Input:aabba**

**Output:a3b2**

Java: public class JavaApplication3 {

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

String str=s.next();

Map<Character,Integer> map = new HashMap<>();

for (int i = 0; i < str.length(); i++) {

char c = str.charAt(i);

if (map.containsKey(c)) {

int cnt = map.get(c);

map.put(c, ++cnt);

} else {

map.put(c, 1);

}

}

System.out.println(map);

}

}

**Output**

aaabbbdj

{a=3, b=3, d=1, j=1}

2. Write a program to sort the elements in odd positions in descending order and elements in ascending order

Eg 1: Input: 13,2 4,15,12,10,5

Output: 13,2,12,10,5,15,4

Eg 2: Input: 1,2,3,4,5,6,7,8,9

Output: 9,2,7,4,5,6,3,8,1

#include<iostream.h>

#include<conio.h>

int main()

{

int n,i,j,temp,\*a=NULL;

cout<<"enetr the size:";

cin>>n;

a=new int[n];

for(i=0;i<n;i++)

{

cin>>a[i];

}

for(i=0;i<n;i=i+2)

{

for(j=i+2;j<n;j=j+2)

{

if(a[i]<a[j])

{

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

}

for(i=0;i<n;i++)

{

cout<<a[i];

}

getch();

return 200;

}

3.cross pattern.

4. Find if a String2 is substring of String1. If it is, return the index of the first occurrence. else return -1.

Eg 1:Input:

String 1: test123string

String 2: 123

Output: 4

Eg 2: Input:

String 1: testing12

String 2: 1234

Output: -1

public class JavaApplication1 {  
   static int check(String str1,String str2)  
    {  
        String str;  
        int a=-1;  
        for(int i=0;i<=str1.length()-str2.length();i++)  
        {  
            str=str1.substring(i,i+str2.length());  
            if(str.equals(str2))  
                return i+1;  
        }  
        return a;  
    }  
    public static void main(String[] args) {  
        Scanner s=new Scanner(System.in);  
        String str1=s.next();  
        String str2=s.next();  
        System.out.println(check(str1,str2));  
          
    }  
      
}

6. Using Recursion reverse the string such as

Eg 1: Input: one two three

Output: three two one

Eg 2: Input: I love india

Output: india love I

#include<stdio.h>

#include <stdlib.h>

int main()

{

int n=0,i=0,s=0,count=0,j,k=0;

char str[100],str1[100];

while(scanf("%c",&str[i])!=-1)

{

n++;

i++;

}

//printf("%d",n);

for(i=n-3;i>=0;i--)

{ count++;

if(str[i]==' '||i==0)

{

for(j=i;j<=count+i;j++)

{

str1[k++]=str[j];

}

count=0;

}

}

for(i=0;i<k;i++)

{

//if(isalpha(str1[i]))

printf("%c",str1[i]);

}

}  
set 4:

1) To find the odd numbers in between the range.  
Input:  
2  
15  
Output:  
3,5,7,9,11,13

Easy prgm do it by urself.!!!! ☺

2) To find the factors of the numbers given in an array and to sort the numbers in descending order according to the factors present in it.  
Input:  
Given array : 8, 2, 3, 12, 16  
Output:  
12, 16, 8, 2, 3

#include<iostream.h>

#include<conio.h>

int main()

{

int \*a=NULL,\*c=NULL,temp,temp1;

int i,j,n;

clrscr();

cout<<"enter the size of the array:";

cin>>n;

a= new int[n];

c=new int[n];

for(i=0;i<n;i++)

{

cin>>a[i];

c[i]=0;

for(j=1;j<=a[i];j++)

{

if(a[i]%j==0)

{ cout<<j<<"\n";

c[i]++;

}

}

}

for(i=0;i<n;i++)

{

for(j=i+1;j<n;j++)

{

if(c[i]<c[j])

{

temp1=c[i];

c[i]=c[j];

c[j]=temp1;

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

}

for(i=0;i<n;i++)

{

cout<<a[i]<<",";

}

getch();

return 200;

}

Output

5

8 12 16 2 3

12 16 8 2 3

3) To output the number in words (0-999)  
Input: 234  
Output: Two hundred and Thirty Four

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication4;

import java.util.Scanner;

/\*\*

\*

\* @author HP

\*/

import java.util.Scanner;

public class JavaApplication4 {

public static void main(String[] args) {

int number ;

Scanner scanner = new Scanner(System.in);

System.out.print("Please type a number between 0 and 999 OR type -1 to exit: ");

number = scanner.nextInt();

while(number!=-1){

if(number>=0 && number<=999){

if(number==0){

System.out.print("NUMBER AFTER CONVERSION:\tZERO");

} else {

System.out.print("NUMBER AFTER CONVERSION:\t");

numberToWord(((number / 100) % 10), " HUNDRED");

numberToWord((number % 100), " ");

}

} else{

System.out.print("NUMBER OUT OF RANGE");

}

System.out.print("\nPlease type a number between 0 and 999 OR type -1 to exit: ");

number = scanner.nextInt();

}

}

public static void numberToWord(int num, String val) {

String ones[] = {" ", " ONE", " TWO", " THREE", " FOUR", " FIVE", " SIX", " SEVEN", " EIGHT", " NINE", " TEN", " ELEVEN", " TWELVE", " THIRTEEN", " FOURTEEN", " FIFTEEN", " SIXTEEN", " SEVENTEEN", " EIGHTEEN", " NINETEEN"

};

String tens[] = {" ", " ", " TWENTY", " THIRTY", " FOURTY", " FIFTY", " SIXTY", " SEVENTY", " EIGHTY", " NINETY"};

if (num > 19) {

System.out.print(tens[num / 10] + " " + ones[num % 10]);

} else {

System.out.print(ones[num]);

}

if (num > 0) {

System.out.print(val);

}

}

}

4) To find the print the pattern:  
Ip: n=5  
Op:  
1  
1 1  
2 1  
1 2 1 1  
1 1 1 2 2 1

????????????????????????????????

5) A man his driving car from home to office with X petrol. There are N number of petrol bunks in the city with only few capacities and each petrol is located in different places For one km one liter will consume. So he fill up petrol in his petrol tank in each petrol bunks. Output the remaining petrol if he has or tell him that he cannot travel if he is out of petrol.  
Input:  
Petrol in car: 2 Liters  
Petrol bunks: A B C  
Distance from petrol each petrol bunks: 1, 5, 3  
Capacities of each petrol bunk: 6, 4, 2  
Output:  
Remaining petrol in car is 5 liters

#include<iostream.h>

#include<conio.h>

int main()

{

int p=2,n,i;

int \*d=NULL,\*c=NULL;

cout<<"enter no of petrol bunks";

cin>>n;

d=new int[n];

c=new int[n];

cout<<"enter distance and capacities";

for(i=0;i<n;i++)

{

cin>>d[i];

cin>>c[i];

p=(p-d[i])+c[i];

}

cout<<"remaining petrol in car"<<p;

getch();

return 200;

}

1) Print the given pattern:

Input:  
N= 3, M=3  
Output:  
X X X  
X 0 X  
X X X

Input:  
N=4 M=5  
Output:  
X X X X  
X 0 0 X  
X 0 0 X  
X 0 0 X  
X X X X

Input:  
N=6 M=7  
X X X X X X  
X 0 0 0 0 X  
X 0 X X 0 X  
X 0 X X 0 X  
X 0 X X 0 X  
X 0 0 0 0 X  
X X X X X X

#include<iostream.h>

#include<conio.h>

int main()

{

int i,j,m,n;

char a[10][10];

clrscr();

cin>>m;

cin>>n;

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

{

a[i][j]='X';

}

}

for(i=1;i<=m-2;i++)

{

for(j=1;j<=n-2;j++)

{

a[i][j]='O';

}

}

for(i=2;i<=m-3;i++)

{

for(j=2;j<=n-3;j++)

{

a[i][j]='X';

}}

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

{

cout<<a[i][j];

}

cout<<"\n";

}

getch();

return 200;

}

1) To output the given string for the given input which is an integer.  
Input: 1  
Output: A  
Input: 26  
Output: Z  
Input : 27  
Output: AA  
Input: 28:  
Output: AB  
Input: 1000  
Output: ALL

#include<iostream.h>  
#include<conio.h>  
int main()  
{  
int n;  
int q,r,a,b;  
char str[26]={'A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z'};  
clrscr();  
cin>>n;  
q=n/26;  
r=n%26;  
if(n<=26)  
{  
 if(n==26)  
cout<<"Z";  
cout<<str[n-1];  
  
}  
if(q<=26 && r<26)  
{  
if(r==0)  
cout<<str[q-1]<<"Z";  
cout<<str[q-1]<<str[r-1];  
}  
else  
{  
a=q/26;  
b=q%26;  
if(b==0)  
cout<<str[a-1]<<"Z"<<str[r-1];  
else if(r==0)  
 cout<<str[a-1]<<str[b-1]<<"Z";  
else if(b==0 &&r==0)  
cout<<str[a-1]<<"Z"<<"Z";  
else  
cout<<str[a-1]<<str[b-1]<<str[r-1];  
}  
getch();  
return 0;  
}

2) Input:  
Number of elements in set1: 4  
Elements are: 9, 9, 9, 9  
Number of elements in set 2: 3  
Elements are: 1,1,1  
Output:  
1, 0, 1, 1, 0  
Input:  
Number of elements in set1: 11  
Elements are: 7,2,3,4,5,3,1,2,7,2,8  
Number of elements in set 2: 3  
Elements are: 1,2,3  
Output: 7,2,3,4,5,3,1,2,8,5,1

#include<iostream.h>

#include<conio.h>

void main()

{

int n,n1,i,j,\*a,\*b,ca=0,\*c,sum;

clrscr();

cin>>n;

cin>>n1;

a=new int[n];

b=new int[n1];

c=new int[n];

for(i=0;i<n;i++)

{

cin>>a[i];

c[i]=a[i];

}

for(i=0;i<n1;i++)

cin>>b[i];

for(i=n-1,j=n1-1;i>=(n-n1-1)&&j>=0;i--,j--)

{

sum=a[i]+b[j]+ca;

if(i==(n-n1))

{

c[i]=sum;

}

else

{

c[i]=sum%10;

ca=sum/10;

}

}

for(i=0;i<n;i++)

{

cout<<c[i];

}

getch();

}

**Set 5:**

**To print the sum of elements with index as power**

**Output**

**3**

**2 3 5**

**29 //2^0+3^1+5^2**

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#include<math.h>

int main()

{

int n,i,sum=0,\*a=NULL;

clrscr();

scanf("%d",&n);

a=(int \*)malloc (sizeof(int) \*n);

for(i=0;i<n;i++)

{

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

sum=sum+(pow(a[i],i));

}

printf("%d",sum);

getch();

return 0;

}

**Program 1:**  
Help john to find new friends in social network  
Input:  
3  
Mani 3 ram raj guna  
Ram 2 kumar Kishore  
Mughil 3 praveen Naveen Ramesh

Output:  
Raj guna kumar Kishore praveen Naveen Ramesh

#include<stdio.h>

#include<conio.h>

#include<string.h>

#define N 10

struct Node{

char name[20];

char list[N][20];

int no;

}friends[N];

void findFriend(int n){

int i,j,k,count,m;

for(i=0;i<n;i++){

m = friends[i].no;

for(j=0;j<m;j++){

for(k=count=0;k<n;k++){

if( strcmpi(friends[i].list[j],friends[k].name) != 0){

count++;

}

}

if(count==n)

printf("%s ",friends[i].list[j]);

}

}

}

int main(){

int t,n,i,j;

clrscr();

scanf("%d",&t);

for(i=0;i<t;i++){

scanf(" %s",friends[i].name);

scanf(" %d",&n);

friends[i].no = n;

for(j=0;j<n;j++){

scanf(" %s",friends[i].list[j]);

}

}

findFriend(t);

getch();

}

Program 2:  
Input:  
With the starting and ending time of work given find the minimum no of workers needed

Start time end time

1230 0130

1200 0100

1600 1700

Output:  
2

#include<iostream.h>

#include<conio.h>

/\* void sort(int a[],int b)

{

int temp;

if(a>b)

{

temp=a;

a=b;

b=temp;

}

} \*/

// Returns minimum number of platforms reqquired

int findworker(int start[], int end[], int n)

{

// Sort arrival and departure arrays

//sort(start,start+n);

//sort(end,end+n);

// plat\_needed indicates number of platforms needed at a time

int worker = 1, result = 1;

int i = 1, j = 0;

// Similar to merge in merge sort to process all events in sorted order

while (i < n && j < n)

{

// If next event in sorted order is arrival, increment count of

// platforms needed

if (start[i] < end[j])

{

worker++;

i++;

if (worker > result) // Update result if needed

result = worker;

}

else // Else decrement count of platforms needed

{

worker--;

j++;

}

}

return result;

}

// Driver program to test methods of graph class

int main()

{

int start[] = {1200, 1400,1600};

int end[] = {1300, 1500, 1700};

int n = sizeof(start)/sizeof(start[0]);

cout << "Minimum Number of worker Required = "

<< findworker(start, end, n);

getch();

return 0;

}

Program 3:  
Find the union intersection of two list and also find except (remove even elements from list1 and odd elements from list2)  
Input

List 1: 1,3,4,5,6,8,9

List 2: 1, 5,8,9,2

Union: 1, 3,4,5,6,8,9,2

Intersection: 1,5,8,9

Except: 1, 3, 5,9,8,2

#include<stdio.h>

#include<conio.h>

void printUnion(int arr1[], int arr2[], int m, int n)

{

int i = 0, j = 0;

while (i < m && j < n)

{

if (arr1[i] < arr2[j])

printf(" %d ", arr1[i++]);

else if (arr2[j] < arr1[i])

printf(" %d ", arr2[j++]);

else

{

printf(" %d ", arr2[j++]);

i++;

}

}

while(i < m)

printf(" %d ", arr1[i++]);

while(j < n)

printf(" %d ", arr2[j++]);

}

void printinter(int arr1[], int arr2[], int m, int n)

{

int i = 0, j = 0;

while (i < m && j < n)

{

if (arr1[i] < arr2[j])

i++;

else if (arr2[j] < arr1[i])

j++;

else

{

printf(" %d ", arr2[j++]);

i++;

}

}

}

void printexcept(int arr1[], int arr2[], int m, int n)

{

int i ;

for(i=0;i < m;i++)

{

if(arr1[i]%2==1)

printf("%d\t",arr1[i]);

}

for(i=0;i <n;i++)

{

if(arr2[i]%2==0)

printf("%d\t",arr2[i]);

}

}

int main()

{

int arr1[] = {1, 2, 4, 5, 6,7,9,10,8};

int arr2[] = {2, 3, 5, 7};

int m = sizeof(arr1)/sizeof(arr1[0]);

int n = sizeof(arr2)/sizeof(arr2[0]);

int i;

clrscr();

printf("array1\n");

for(i=0;i<m;i++)

printf("%d\t",arr1[i]);

printf("\narray2\n");

for(i=0;i<n;i++)

printf("%d\t",arr2[i]);

printf("\nunion:\n");

printUnion(arr1, arr2, m, n);

printf("\nintersection\n");

printinter(arr1,arr2,m,n);

printf("\nexcept\n");

printexcept(arr1,arr2,m,n);

getchar();

return 0;

}

Program 4:

Rotate the matrix elements

For 3\*3 matrix

Input

1 2 3

4 5 6

7 8 9

Output:

4 1 2

7 5 3

8 9 6

For 4\*4 matrix

Input:

1 2 3 4

5 6 7 8

9 10 11 12

13 14 15 16

Output:

5 1 2 3

9 10 6 4

13 11 7 8

14 15 16 12

**Rotate matrix anticlockwise**

#include<iostream.h>

#include<conio.h>

int main()

{

int a[3][3],temp,i,j,n=4;

clrscr();

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

cin>>a[i][j];

}

}

for(i=0;i<n/2;i++)

{

for(j=i;j<(n-i-1);j++)

{

int temp=a[i][j];

a[i][j]=a[j][n-1-i];

a[j][n-1-i]=a[n-1-i][n-1-j];

a[n-1-i][n-1-j]=a[n-1-j][i];

a[n-1-j][i]=temp;

}

}

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

cout<<a[i][j];

}

cout<<"\n";

}

getch();

return 0;

}

**Rotate matrix clockwise**

#include<iostream.h>

#include<conio.h>

void roll(int &a,int &b,int &c,int &d)

{

int temp=a;

a=b;

b=c;

c=d;

d=temp;

}

int main()

{

int a[3][3],temp,i,j,n=4;

clrscr();

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

cin>>a[i][j];

}

}

for(i=0;i<n/2;i++)

{

for(int j=0; j<(n+1)/2; j++)

cyclic\_roll(m[i][j], m[n-1-j][i], m[n-1-i][n-1-j], m[j][n-1-i]);

}

}

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

cout<<a[i][j];

}

cout<<"\n";

}

getch();

return 0;

}

**Set 6**

Q1. Given dates in day,month, year order sort them.

#include<stdio.h>

#include<conio.h>

void yexch(int y[],int m[],int d[],int i, int j)

{

int temp,temp1,temp2;

temp=y[i];

y[i]=y[j];

y[j]=temp;

temp1=m[i];

m[i]=m[j];

m[j]=temp1;

temp2=d[i];

d[i]=d[j];

d[j]=temp2;

}

int main()

{

int d[10];

int y[10];

int m[10],n,i,j;

clrscr();

printf("enter no of dates");

scanf("%d",&n);

for(i=0;i<n;i++)

scanf("%d%d%d" ,&d[i],&m[i],&y[i]);

for(i=0;i<n;i++)

{

for(j=i+1;j<n;j++)

{

if(y[i]>y[j])

{

yexch(y,m,d,i,j);

}

else if(y[i]==y[j])

{

if(m[i]>m[j])

{

yexch(y,m,d,i,j);

}

else if(m[i]==m[j])

{

if(d[i]>d[j])

yexch(y,m,d,i,j);

}

else

{

}

}

else

{

}

}}

for(i=0;i<n;i++)

{

printf("\n%d %d %d",d[i],m[i],y[i]);

}

getch();

return 200;

}

.

Q3: Find whether a given number is magic number or not. It is something which gives same digits even after cubing it.eg:1729 12^3+1^3=1729 (or) 1+7+2+9=19 => 1+9=10 => 1+0=1

A number is said to be a Magic number if the sum of its digits are calculated till a single digit is obtained by recursively adding the sum of its digits.  
If the single digit comes to be 1 then the number is a magic number.  
Example- 199 is a magic number as 1+9+9=19 but 19 is not a single digit number so 1+9=10 and then 1+0=1 which is a single digit number and also 1.Hence it is a magic number.

Kavi’s solution:

#include<iostream.h>

#include<conio.h>

void rem(int n)

{

int temp=n;

int sum=0;

int dig;

while(temp>0)

{

dig=temp%10;

sum=sum+dig;

temp=temp/10;

}

if(sum>=10)

rem(sum);

else if(sum==1)

cout<<"magic";

else

cout<<"no magic";

}

int main()

{

int n;

cin>>n;

rem(n);

getch();

return 200;

}

My solution:

#include<iostream.h>

#include<conio.h>

int function(int);

int rem,add=0;

int main()

{

float n;

int b;

clrscr();

cout<<"enter a number:";

cin>>n;

cout<<"\nadd:"<<function(n);

if(add==1)

{

cout<<"\nmagic number";

}

else

{

cout<<"\nnot a magic number";

}

getch();

return 200;

}

int function(int n)

{

add=0;

int a=0;

while(n>0)

{

rem=n%10;

//sum=sum\*10+rem;

n=n/10;

add=add+rem;

//cout<<"\nrem:"<<rem;

a++;

}

//cout<<"add"<<add;

//cout<<"sizeof:"<<a;

if(a!=1)

{

function(add);

}

return add;

}  
Q2. Given a string of integers find out all the possible words that can made out of it in continuous order. Eg: 11112

ans:

AAAAB  
AKAB  
AAKB  
AAAL etc.

public class JavaApplication5 {  
    public static void main(String[] args) {  
      x("11112", "");  
}  
private static final void x(String digits, String word) {  
    if (digits.isEmpty())  
        System.out.println(word);  
    else {  
        int num = Integer.parseInt(digits.substring(0, 1));  
        x(digits.substring(1), word + (char)('A' + num - 1));  
        if (digits.length() >= 2 && (num = Integer.parseInt(digits.substring(0, 2))) <= 26)  
            x(digits.substring(2), word + (char)('A' + num - 1));  
    }  
}  
}

Q5: Given two numbers and an operation either + or – , perform the operation.  
Now remove any zeros if present in the two numbers and perform an operation. See if the result obtained is same or not after removing zero’s in the original result. terla

SET 7:

1.Cyclic number verification terla  
2. Sorting dates already done  
3. write a code to solve given mathematical expression terla  
4. generation of unique number from any random number down  
5. given a number u need to print all combination of alphabets for that number already done???????????????????????????????????

4. you checked most basic random number generation program in C, you might have some knowledge about rand() function in C which is used to generate random numbers.  
  
Why I said that most basic?  
Because, rand() function works very well but after running that program again and again you can identify that its generating the same series or number all the time.  
  
So how to fix?  
To fix this common issue we usually need to seed our rand function using srand() function. And this is how srand function is used to generate different number all the time:

srand(time(NULL));

So now we know how to generate random number and how to fix its common problem. Now we are all set to define our range and actually generate random number in that range which will not repeat itself again.

#include <iostream.h>

#include<conio.h>

#include<stdlib.h>

void main(){

int len ,i , r , temp,N1,N2;

int \*num=new int[len];

clrscr();

cout<<"enter range between to print random numbers";

cin>>N1;

cin>>N2;

len=N2-N1+1;

//Fill array with desired numbers

for( temp=0,i=N1; temp<len; i++,temp++ )

num[temp] = i;

srand( time(NULL) ); //seed rand()

//Fisher–Yates shuffle algorithm

for( i=len-1; i>0; i-- ){

r = rand()%i; //pop random number

//swaping using temp

temp = num[i];

num[i] = num[r];

num[r] = temp;

}

/\*Random Numbers b/w N1-N2 are stored in Array num\*/

//print the array

for( i=0; i<len; i++ )

cout<<"\n"<<num[i];

getch();

}

Output

Enter a range for random numbers 4 8

7

8

5

6

4

SET 8:

1. Spiral printing.

O/P

4444444

4333334

4322234

4321234

4322234

4333334

4444444

#include<iostream.h>

#include<conio.h>

void set(int m,int n,int i,int m1)

{

int a[20][20];

int f,g;

for(f=i;f<m;f++)

{

for(g=i;g<m;g++)

{

a[f][g]=n;

}

}

if(n==1)

{

for(f=0;f<m1;f++)

{

for(g=0;g<m1;g++)

{

cout<<a[f][g];

}

cout<<"\n";

}

}

}

int main()

{

int a[10][10];

int i,j=0,m,n;

clrscr();

cin>>m;

int t=(m/2)+1;

n=t;

int l=m;

for(i=0;i<t;i++)

{

set(l,n,j,m);

n=n-1;

l=l-1;

j=j+1;

}

getch();

return 200;

}

2. Sort the array alternately i.e first element should be max value, second min value, third second max, third second min. Eg: arr[] = {1,2,3,4,5,6,7} O/P: {7,1,6,2,5,3,4} Note: no extra space and time complexity should be less; already done

3. Print all the substring of the given string. (net but puriyum)

/ C program to print all permutations with duplicates allowed

#include<stdio.h>

#include<string.h>

#include<conio.h>

/\* Function to swap values at two pointers \*/

void swap(char \*x, char \*y)

{

char temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

/\* Function to print permutations of string

This function takes three parameters:

1. String

2. Starting index of the string

3. Ending index of the string. \*/

void permute(char \*a, int l, int r)

{

int i;

printf("%s\t",a);

if (l == r)

printf("%s\n", a);

else

{

for (i = l; i <= r; i++)

{

swap((a+l), (a+i));

permute(a, l+1, r);

swap((a+l), (a+i)); //backtrack

}

}

}

/\* Driver program to test above functions \*/

int main()

{

char str[50] ;

int n;

clrscr();

scanf("%s",&str);

n = strlen(str);

permute(str, 0, n-1);

getch();

return 0;

}

4. Print the numbers which are mismatched from two array. Arr1 = {a b c d e f g h i}  
arr2 ={ a b d e e g g i i}, O/P- cd, de, f, g, h, i.

#include<iostream.h>

#include<conio.h>

int main()

{

char a[10],b[10];

int i,n;

clrscr();

cin>>n;

cout<<"\nenter first array";

for(i=0;i<n;i++)

{

cin>>a[i];

}

cout<<"\nenter second array";

for(i=0;i<n;i++)

{

cin>>b[i];

}

cout<<"\n";

for(i=0;i<n;i++)

{

if(a[i]!=b[i])

{

cout<<a[i]<<b[i];

cout<<",";

}

}

getch();

return 200;

}

5. Print all possible combinations from the given string.(net ) puriyadhu light ah

public class JavaApplication6 {

public static void main(String[] args) {

String string, sub;

int i, c, length;

Scanner in = new Scanner(System.in);

System.out.println("Enter a string to print it's all substrings");

string = in.nextLine();

length = string.length();

System.out.println("Substrings of \""+string+"\" are :-");

/\* for( c = 0 ; c < length ; c++ )

{

for( i = 1 ; i <= length - c ; i++ )

{

sub = string.substring(c, c+i);

System.out.println(sub);

}

}\*/

for (i = 0; i < string.length(); i++) {

for (int j = i+1; j <= string.length(); j++) {

System.out.println(string.substring(i,j));

}

}