**Simple Array Sum**

#include<iostream>

using namespace std;

int main()

{

    int n,i;

    cin>>n;

    int ar[n];

    int t=0;

    for(i=0;i<n;i++)

    {

        cin>>ar[i];

        t+=ar[i];

    }

    cout<<t;

}

**A Very Big Sum**

#include<iostream>

using namespace std;

int main()

{

    int n,i;

    cin>>n;

    long ar[n],t=0;

    for(i=0;i<n;i++)

    {

        cin>>ar[i];

        t+=ar[i];

    }

    cout<<t;

}

**Diagonal Difference**

#include<iostream>

#include<math.h>

using namespace std;

int main()

{

    int n,i,j;

    cin>>n;

    int ar[n][n];

    for(i=0;i<n;i++)

    {

        for(j=0;j<n;j++)

        {

            cin>>ar[i][j];

        }

    }

    int t=0;

    int y=0;

    for(i=0;i<n;i++)

    {

        t+=ar[i][i];

        y+=ar[i][n-i-1];

    }

    int r=abs(t-y);

    cout<<r;

}

**Equalize the Array**

#include<iostream>

using namespace std;

int main()

{

    int n,i,j;

    cin>>n;

    int ar[n];

    for(i=0;i<n;i++)

    {

        cin>>ar[i];

    }

    int max=0;

    int t=0;

    for(i=0;i<n;i++)

    {

        t=0;

        for(j=0;j<n;j++)

        {

            if(ar[i]==ar[j])

                t++;

        }

        if(t>max)

        {

            max=t;

        }

    }

    cout<<n-max;

}

**Mini-Max Sum**

#include<iostream>

using namespace std;

int main()

{

    long ar[5];

    int i;

    long t=0;

    long min=10000000000000,max=0;

    for(i=0;i<5;i++)

    {

        cin>>ar[i];

    }

    for(i=0;i<5;i++)

    {

        t+=ar[i];

        if(ar[i]>max)

        {

            max=ar[i];

        }

        if(ar[i]<min)

        {

            min=ar[i];

        }

    }

    long s=t-max;

    long f=t-min;

    cout<<s<<" "<<f;

}

**Extra Long Factorials**

#include<iostream>

using namespace std;

int factorial(int x,int ar[],int size)

{

    int carry=0;

    int i;

    for(i=0;i<size;i++)

    {

        int p=ar[i]\*x+carry;

        ar[i]=p%10;

        carry=p/10;

    }

    while(carry)

    {

        ar[size]=carry%10;

        carry=carry/10;

        size++;

    }

    return size;

}

int main()

{

    int n,i;

    cin>>n;

    int ar[500];

    int size=1;

    ar[0]=1;

    for(i=2;i<=n;i++)

    {

        size=factorial(i,ar,size);

    }

    for(i=size-1;i>=0;i--)

    {

        cout<<ar[i];

    }

}

**Kangaroo**

#include<iostream>

using namespace std;

int main()

{

    int x1,v1,x2,v2;

    cin>>x1>>v1>>x2>>v2;

    int r=0;

    while((x1<x2&&v1>v2&&r==0)||(x1>x2&&v1<v2&&r==0))

    {

        x1+=v1;

        x2+=v2;

        if(x1==x2)

        {

            r=1;

        }

    }

    if(r==1)

    {

        cout<<"YES";

    }

    else

    {

        cout<<"NO";

    }

}

**Halloween Sale**

#include<iostream>

#include<math.h>

using namespace std;

int main()

{

    int p,d,m,s;

    cin>>p;

    cin>>d;

    cin>>m;

    cin>>s;

    int a=0;

    if(m<=p)

    {

    while(s>=0)

    {

        s=s-max(p,m);

        p=p-d;

        a++;

    }}

    int g=max(0,a-1);

    cout<<g;

    return 0;

**}**

**Non-Divisible Subset**

#include<iostream>

#include<math.h>

using namespace std;

int main()

{

    int n,k,i;

    cin>>n>>k;

    int ar[n];

    for(i=0;i<n;i++)

    {

        cin>>ar[i];

    }

    int b[k];

    for(i=0;i<k;i++)

    {

        b[i]=0;

    }

    for(i=0;i<n;i++)

    {

        b[ar[i]%k]+=1;

    }

    int c;

    c=min(1,b[0]);

    for(i=1;i<k/2+1;i++)

    {

        if(i!=k-i)

        {

            c+=max(b[i],b[k-i]);

        }

    }

    if(k%2==0)

    {

        c++;

    }

    cout<<c;

}

**Forming a Magic Square**

#include<iostream>

#include<math.h>

using namespace std;

int main()

{

    int ar[3][3],i,j;

    for(i=0;i<3;i++)

    {

        for(j=0;j<3;j++)

        {

            cin>>ar[i][j];

        }

    }

    int abr[8][3][3]={{{8,1,6},{3,5,7},{4,9,2}},{{6,1,8},{7,5,3},{2,9,4}},{{4,9,2},{3,5,7},{8,1,6}},{{2,9,4},{7,5,3},{6,1,8}},{{8,3,4},{1,5,9},{6,7,2}},{{4,3,8},{9,5,1},{2,7,6}},{{6,7,2},{1,5,9},{8,3,4}},{{2,7,6},{9,5,1},{4,3,8}}};

    int minn=10000;

    int k;

    for(k=0;k<8;k++)

    {

        int cost=0;

        for(i=0;i<3;i++)

        {

            for(j=0;j<3;j++)

            {

                cost+=abs(ar[i][j]-abr[k][i][j]);

            }

        }

        minn=min(minn,cost);

    }

    cout<<minn;

}