

От първия ред на стандартния вход програмата трябва да прочете стойностите на M , N , m и n . Следват числата от дадената таблица, записани последователно ред по ред. Всички числа във входа са разделени с интервали или със символи за край на ред. Програмата трябва да изведе резултата на стандартния изход.

ПРИМЕР

Вход

```
4 6 2 2
1 -1 2 3 4 -2
-3 4 -5 6 -1 0
3 -1 -2 1 4 -2
-2 4 -5 4 -1 1
```

Изход

```
12
```

РЕШЕНИЕ

Елементите на дадената таблица се въвеждат като елементи на масив $a[i][j]$. За всеки елемент на таблицата с индекси i и j се пресмятат като елементи на нов масив $b[i][j]$ сумата от стойностите на числата в клетките от първия масив, намиращи се в подтаблица, съставена от последователни редове с номера от 1 до i , и последователни стълбове с номера от 1 до j . За бързо пресмятане са използвани рекурентни формули (Възможно е използването на същия масив a за запазване на тези суми, без да се въвежда масив b). След това се обхождат всички клетки на новата таблица $b[i][j]$. За всяка клетка се пресмята сумата от елементите на разглежданите подтаблицы според условието на задачата (отново чрез използване на рекурентни формули) и едновременно се намира максимумът res от всичките пресметнати суми.

```
#include<cstdio>
#include<climits>
#include<algorithm>
using namespace std;

const int M_max=1001;
const int N_max=1001;
int a[M_max][N_max], b[M_max][N_max];
int M,N,m,n;

int main()
{
    scanf("%d %d %d %d", &M,&N,&m,&n);
    for(int i=1;i<=M;i++)
        for(int j=1;j<=N;j++) scanf("%ld", &a[i][j]);

    b[1][1]=a[1][1];
    for(int j=2;j<=N;j++) b[1][j]=b[1][j-1]+a[1][j];
    for(int i=2;i<=M;i++) b[i][1]=b[i-1][1]+a[i][1];

    for(int i=2;i<=M;i++)
        for(int j=2;j<=N;j++)
            b[i][j]=a[i][j]+b[i-1][j]+b[i][j-1]-b[i-1][j-1];
```

```

int res=LONG_MIN;
for(int i=1;i<=M;i++)
for(int j=1;j<=N;j++)
    if((i<m)|| (j<n))
    {}
    else if((i==m)&&(j==n))
        {int v=b[i][j];res=max(res,v);}
    else if((i==m)&&(j>n))
        {int v=b[i][j]-b[i][j-n];res=max(res,v);}
    else if((i>m)&&(j==n))
        {int v= b[i][j]-b[i-m][j];res=max(res,v);}
    else
        {int v=b[i][j]+b[i-m][j-n]-b[i-m][j]-b[i][j-n];res=max(res,v);}
printf("%d\n",res);
}

```

Емил Келеведжиев