

#5 铺地毯 解答

代码

```
#include <iostream>
#include <vector>

using namespace std;

struct Point
{
    int x, y;
};

struct Area
{
    Point v;
    int w, h;
};

bool in(Point p, Area a)
{
    int dx = p.x - a.v.x;
    int dy = p.y - a.v.y;
    return (dx >= 0 && dx <= a.w) && (dy >= 0 && dy <= a.h);
}

int main()
{
    int n;
    vector<Area> arr;
    Point p;
    cin >> n;
    for (int i = 0; i < n; i++) {
        Area a;
        cin >> a.v.x >> a.v.y >> a.w >> a.h;
        arr.push_back(a);
    }
    cin >> p.x >> p.y;
    for (int i = arr.size()-1; i >= 0; i--) {
        if (in(p, arr[i])) {
            cout << i+1;
            return 0;
        }
    }
}
```

```
    cout << -1;  
  
    return 0;  
}
```

解析

这是一道很好的数组样例。

由于毯子的输入顺序是从下到上，要获取最上方的毯子，只需现将数据保存下来，再从后往前（也即从上到下）考察每一张毯子，判断那个点是否在毯子的范围内即可。