

## 1233: 接水问题

## 题目描述

学校里有一个水房，水房里一共装有 $m$ 个龙头可供同学们打开水，每个龙头每秒钟的供水量相等，均为1。

现在有 $n$ 名同学准备接水，他们的初始接水顺序已经确定。将这些同学按接水顺序从1到 $n$ 编号， $i$ 号同学的接水量为 $w_i$ 。接水开始时，1到 $m$ 号同学各占一个水龙头，并同时打开水龙头接水。当其中某名同学 $j$ 完成其接水量要求 $w_j$ 后，下一名排队等候接水的同学 $k$ 马上接替 $j$ 同学的位置开始接水。这个换人的过程是瞬间完成的，且没有任何水的浪费。即 $j$ 同学第 $x$ 秒结束时完成接水，则 $k$ 同学第 $x+1$ 秒立刻开始接水。若当前接水人数 $n'$ 不足 $m$ ，则只有 $n'$ 个龙头供水，其它 $m-n'$ 个龙头关闭。

现在给出 $n$ 名同学的接水量，按照上述接水规则，问所有同学都接完水需要多少秒。

## 输入

第1行2个整数 $n$ 和 $m$ ，用一个空格隔开，分别表示接水人数和龙头个数。

第2行 $n$ 个整数  $w_1$ 、 $w_2$ 、……、 $w_n$ ，每两个整数之间用一个空格隔开， $w_i$ 表示  $i$  号同学的接水量。

## 输出

输出只有一行，1个整数，表示接水所需的总时间。

## 输入样例

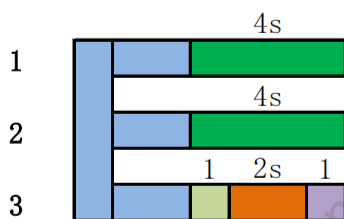
```
5 3
4 4 1 2 1
```

## 输出样例

```
4
```

## 解析

所有人都是按照输入的顺序，先后的排队接水。对于总时间的大小，决定因素是接水的人选择哪个水龙头。如图所示：



通过图上分析，我们知道，新进来的接水者必然会选择总时间最短的那个水龙头，因此，我们只需要不断的对已经在水龙头接水的人进行排序，即可得到最短的接水时间。

## 编码

```
#include<bits/stdc++.h>

using namespace std;
int n, m, b;
//水龙头的数组
int a[100];

int main() {
    //读入n个人和m个水龙头
    cin >> n >> m;
    //接下来读入全部的接水时间
    for (int i = 0; i < n; i++) {
        cin >> b;
        //对m个水龙头进行排序，注意左闭右开
        //sort默认是升序，所以a[0]为最小值
        sort(a, a + m);
        //将下一位同学的接水时间累加到a[0]中
        a[0] += b;
    }
    //最后进行一次排序，取出全局最大值，最大值就是a[m-1];
    sort(a, a + m);
    cout << a[m - 1];
    return 0;
}
```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

