

题目描述：
有 N 块二手市场收集的银饰，每块银饰的重量都是正整数，收集到的银饰会被熔化用于打造新的饰品。
每一回合，从中选出三块 最重的 银饰，然后一起熔掉。假设银饰的重量分别为 x 、 y 和 z ，且 $x \leq y \leq z$ 。那么熔掉的可能结果如下：
如果 $x = y = z$ ，那么三块银饰都会被完全熔掉；
如果 $x = y$ 且 $y \neq z$ ，会剩余重量为 $z - y$ 的银块无法被熔掉；
如果 $x \neq y$ 且 $y = z$ ，会剩余重量为 $y - x$ 的银块无法被熔掉；
如果 $x \neq y$ 且 $y \neq z$ ，会剩余重量为 $z - y$ 与 $y - x$ 差值 的银块无法被熔掉。
最后，如果剩余两块，返回较大的重量（若两块重量相同，返回任意一块皆可）；如果只剩下一块，返回该块的重量；如果没有剩下，就返回 0。
输入描述：
输入数据为两行
第一行为银饰数组长度 n ， $1 \leq n \leq 40$ ，
第二行为 n 块银饰的重量，重量的取值范围为 $[1, 2000]$ ，重量之间使用空格隔开
输出描述：
如果剩余两块，返回较大的重量（若两块重量相同，返回任意一块皆可）；如果只剩下一块，返回该块的重量；如果没有剩下，就返回 0。
补充说明：
示例 1
输入：

3
1 1 1

输出：
0

说明：

选出 1 1 1，得到 0，最终数组转换为 []，最后没有剩下银块，返回 0

示例 2

输入：

3
3 7 10

输出:

1

说明:

选出 3 7 10, 需要计算 $(7-3)$ 和 $(10-7)$ 的差值, 即 $(7-3)-(10-7)=1$, 所以数组转换为 [1], 剩余一块, 返回该块重量, 返回 1

```
#include <algorithm>
#include <iostream>
#include <list>
using namespace std;

void print( std::list<int>& ls) {
    for (auto& it : ls) {
        std::cout << it << " ";
    }
    std::cout << std::endl;
}

int main() {
    std::list<int> ls;
    int tmp;
    std::cin >> tmp;
    while (std::cin >> tmp)
        ls.emplace_back(tmp);
    // print(ls);
    while (ls.size() > 2) {
        ls.sort([](auto & _1, auto & _2) {
            return _2 < _1;
        });
        auto it = ls.begin();
        auto& _1 = *it++;
        auto& _2 = *it++;
        auto& _3 = *it++;
        if ( _1 == _2 && _2 == _3) {
            ls.erase(ls.begin(), it);
            ls.erase(ls.begin(), it);
            ls.erase(ls.begin(), it);
            // std::cout << "all same" << std::endl;
            // print(ls);
            continue;
        }
        if ( _3 == _2 && _2 != _1) { //y==z x!=y
            _3 = _1 - _2;
```

```

        ls.erase(ls.begin());
        ls.erase(ls.begin());
        // std::cout << "y=z, x!=y" << std::endl;
        // print(ls);
        continue;
    }
    if (_3 != _2 && _2 == _1) {
        _3 = _2 - _3;
        ls.erase(ls.begin());
        ls.erase(ls.begin());
        // std::cout << "y=x, z!=y" << std::endl;
        // print(ls);
        continue;
    }
    // std::cout << "y!=z, x!=y" << std::endl;
    // print(ls);
    _3 = std::max(_2 - _3, _1 - _2) - std::min(_2 - _3, _1 - _2);
    if(_3 == 0)
        ls.erase(ls.begin());
        ls.erase(ls.begin());
        ls.erase(ls.begin());
    // print(ls);
    continue;
}
if (ls.empty())
    std::cout << 0 << std::endl;
else if (ls.size() == 1)
    std::cout << *ls.begin() << std::endl;
else std::cout << std::max(*ls.begin(), *ls.rbegin());
return 0;
}
// 64 位输出请用 printf("%lld")

```