

给一个数组，如何同时求出最大值和最小值，如何同时求出最大值和第二大值？

基础解法：

```
std::vector<int> max_and_min(std::vector<int> nums) {
    int max = INT_MIN;
    int min = INT_MAX;
    for(int i = 0; i < nums.size(); i++) {
        if(nums[i] >= max) {
            max = nums[i];
        }

        if(nums[i] <= min) {
            min = nums[i];
        }
    }

    return std::vector<int>{max, min};
}

std::vector<int> max_and_max_2(std::vector<int> nums) {
    int max = INT_MIN;
    int max2 = INT_MIN;
    for(int i = 0; i < nums.size(); i++) {
        if(nums[i] >= max) {
            max2 = max;
            max = nums[i];
        } else if(nums[i] >= max2) {
            max2 = nums[i];
        }
    }

    return std::vector<int>{max, max2};
}
```

对max\_and\_min可以每次取两个元素，两者较大的与max,较小的与min比较：

```
std::vector<int> max_and_min(std::vector<int> nums) {
    int max = INT_MIN;
    int min = INT_MAX;
    for(int i = 0; i < nums.size(); i+=2) {
        if(nums[i] >= nums[i+1]) {
            max = std::max(max, nums[i]);
            min = std::min(min, nums[i+1]);
        } else {
            max = std::max(max, nums[i+1]);
            min = std::min(min, nums[i]);
        }
    }
}
```

```

    }
}

return std::vector<int>{max, min};
}

```

对max\_and\_max2采用分治法:

```

std::vector<int> max1_and_max2(std::vector<int> nums) {
    return divide_conquer(nums, 0, nums.length);
}

// 返回集合 A 在索引区间 [i, j) 的最大元素和第二大元素
std::vector<int> divide_conquer(std::vector<int> A, int i, int j) {
    // base case
    if (j - i == 2) {
        if (A[i] < A[i + 1]) {
            return std::vector<int>{A[i + 1], A[i]};
        } else {
            return std::vector<int>{A[i], A[i + 1]};
        }
    }
    // 解决两个子问题
    std::vector<int> left = divide_conquer(A, i, (i + j) / 2);
    std::vector<int> right = divide_conquer(A, (i + j) / 2, j);

    int p1 = left[0], p2 = left[1];
    int q1 = right[0], q2 = right[1];
    // 通过子问题的结果合并原问题的答案
    int max1, max2;
    if (p1 > q1) {
        max1 = p1;
        max2 = Math.max(p2, q1);
    } else {
        max1 = q1;
        max2 = Math.max(q2, p1);
    }
    return std::vector<int>{max1, max2};
}

```