



2

A horizontal row of fifteen empty square boxes, intended for children to draw or color in.

2

三

1

1

- မြန်မာစာ မျက်
 - မြန်မာစာ မျက်မျက်

1

set□□□□□□□□□

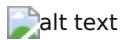
set 

set █ █ █ █

- `vector` `set` `set<int> s;`
 - `s.insert(3);` `vector`
 - `set` `pair`
 - `vector` `vector` `pair` `vector`
 - `set<pair<int,int>>`
 - `vector` `vector` `vector` `vector`
 - `for(auto x: s) x` `vector` `s` `vector`
 - `vector` `pair` `vector` `vector`
 - `x.first` `vector` `vector`
 - `x.second` `vector` `vector`
 - `s.begin()->first` `vector`

- `s.begin()` `s.end()`
- `s.begin() -> s.end()`
- `s.erase()` `s.clear()`
- `s.size()` `s.empty()`

二分查找K题

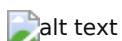


- `K题 -> 二分查找`
- `K题 -> 有序数组`
- `K题 -> 有序数组二分查找`
- `K题 -> 有序数组二分查找`

```
class KthLargest {
public:
    typedef pair<int,int> PII;
    int tot,k;
    set<PII> s;
    KthLargest(int k, vector<int>& nums) {
        this->k = k; //this->k = k;
        for(auto x: nums)
        {
            add(x);
        }
        return;
    }

    int add(int val) {
        if(s.size() < k)
        {
            s.insert(PII(val,tot++));
        }
        else{
            if(s.begin()->first < val)
            {
                s.insert(PII(val,tot++));
            }
        }
        if(s.size() > k) s.erase(s.begin());
        return s.begin()->first;
    }
};
```

二分查找进阶



- `二分查找进阶`
- `二分查找进阶`

- 亂數
- 亂數產生器
- 亂數產生器的實現
- 亂數產生器的應用
- 亂數產生器的應用
 - 亂數產生器
 - 亂數產生器
- 亂數產生器的實現1
- **n1**亂數產生器的實現 **n1**亂數產生器的實現

```

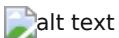
class MedianFinder {
public:
    typedef pair<int,int> PII;
    int tot;
    set<PII> s1,s2;
    MedianFinder() {
        tot = 0;
    }

    void addNum(int num) {
        if(s1.size() == 0 || num < -s1.begin()->first)
            s1.insert(PII(-num,tot++));
        else
            s2.insert(PII(num,tot++));
        int n1 = (s1.size() + s2.size() + 1) / 2;
        if(n1 == s1.size()) return;
        if(s1.size() < n1)
        {
            s1.insert(PII(-s2.begin()->first,tot++));
            s2.erase(s2.begin());
        }
        else
        {
            s2.insert(PII(-s1.begin()->first,tot++));
            s1.erase(s1.begin());
        }
    }

    double findMedian() {
        if((s1.size() + s2.size()) % 2)
            return -s1.begin()->first;
        double a = -s1.begin()->first;
        double b = s2.begin()->first;
        return (a + b) / 2.0;
    }
};

```

亂數產生器



- 00000000000000000000000000000000
 - 00000000000000000000000000000000
 - 00000000000000000000000000000000
 - 00000000000000000000000000000000

```
class Solution {
public:
    ListNode* mergeKLists(vector<ListNode*>& lists) {
        typedef pair<int,int> PII;
        set<PII> s;
        int n = lists.size();
        for(int i = 0;i < n;i++)
        {
            if(lists[i] == nullptr) continue;
            s.insert(PII(lists[i]->val,i));
        }
        ListNode new_head,*p = &new_head , *q;
        new_head.next = nullptr;
        while(s.size())
        {
            PII a = *s.begin();
            s.erase(s.begin());
            q = lists[a.second];
            lists[a.second] = lists[a.second]->next;
            p->next = q;
            q->next = nullptr;
            p = q;
            if(lists[a.second])
            {
                s.insert(PII(lists[a.second]->val,a.second));
            }
        }
        return new_head.next;
    }
};
```

2



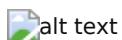
- 1
 - (1)(2)(3)(4)(5)
 - nans nans nans alt text
 - (2)(3)(5)

```

class Solution {
public:
    int nthUglyNumber(int n) {
        set<long long> s;
        s.insert(1);
        long long ans = 0;
        while(n--)
        {
            ans = *s.begin();
            s.erase(s.begin());
            if(ans % 5 == 0)
            {
                s.insert(ans * 5);
            }
            else if(ans % 3 == 0)
            {
                s.insert(ans * 3);
                s.insert(ans * 5);
            }
            else
            {
                s.insert(ans * 2);
                s.insert(ans * 3);
                s.insert(ans * 5);
            }
        }
        return ans;
    }
};

```

解説



- 丑数の定義
- 丑数の性質
- 丑数の生成方法
 - 素因数分解
 - 丑数の倍数を生成する
 - 丑数の倍数を生成する
- c++による実装 Data構造体

```

#include<bits/stdc++.h>
using namespace std;
struct Data
{
    int p,d;
    Data(int p,int d):p(p),d(d){}
}

```

```

        bool operator<(const Data &obj) const
    {
        if(d != obj.d) return d < obj.d;
        return p > obj.p;
    }
};

typedef pair<int,int> PII;

int main()
{
    int n;
    cin >> n;
    vector<Data> arr;
    set<PII> s;
    for(int i = 0,p,d;i < n;i++)
    {
        cin >> p >> d;
        arr.push_back(Data(p,d));
    }
    sort(arr.begin(),arr.end());
    for(int i = 0;i < n;i++)
    {
        if(arr[i].d > s.size())
            s.insert(PII(arr[i].p,i));
        else
        {
            if(arr[i].p > s.begin()->first)
            {
                s.erase(s.begin());
                s.insert(PII(arr[i].p,i));
            }
        }
    }
    int ans = 0;
    for(auto x: s)
        ans += x.first;
    cout << ans;
    return 0;
}

```

解題メモ

M-1

```

#include<bits/stdc++.h>
using namespace std;
typedef pair<int,int> PII;
int main()
{

```

```
int n,m,t = 0;
cin >> n >> m;
set<PII> s;
s.insert(PII(0, t++));
for(int i = 0;i < n;i++)
{
    vector<int> temp;
    for(auto x : s)
    {
        temp.push_back(x.first);
    }
    s.clear();
    for(int j = 0 ,a;j < m;j++)
    {
        cin >> a;
        for(auto x: temp)
        {
            if(s.size() < m || s.begin()->first < x - a)//删除
                s.insert(PII(x - a,t++));
            if(s.size() > m) s.erase(s.begin());
        }
    }
}
int flag = 0;
for(auto iter = s.rbegin();iter != s.rend();iter++)//输出
{
    if(flag) cout << " ";
    cout << -iter->first ;
    flag = 1;
}
return 0;
}
```