

—:

- B: $\sum_{i=0}^{127} i = 63.5$

==:

- B

- ```
void Union(int n,int* A,int* B,int* C){
 int i=0,j=0,k=0;
 while(i<n&& j<n){
 if(A[i]<B[j])C[k++]=A[i++];
 else C[k++]=B[j++];
 }
 while(i<n)C[k++]=A[i++];
 while(j<n)C[k++]=B[j++];
}
```

≡:

- ```
#include<bits/stdc++.h>
using namespace std;
template<class T>class node;
template<class T>ostream& operator<<(ostream& os,const
node<T>& _);
template<class S,class T>void Delete(int& n,S A,const T&
item);
template<class T>class node{
public:
    T val;
    node* next;
    node(){}
    node(T val):val(val),next(NULL){}
    bool operator !=(const T& _)const{return next->val!=_;}
    friend ostream& operator<< <T>(ostream& os,const node<T>&
_);
    node& operator =(const node& _){return next=_>next,*this;}
};
```

```

template<class T>ostream& operator<<(ostream& os,const
node<T>& _){return os<<_.next->val;}
template<class T>class node_iterator{
public:
    node<T>* p;
    node_iterator(node<T>* p):p(p){}
    node_iterator& operator ++(int){
        p=p->next;
        return *this;
    }
    void Link(T v){
        p->next=new node<T>(v);
        p=p->next;
    }
    node<T>& operator *()const{return *p;}
};
template<class S,class T>void TEST(int& n,S A,const T& item){
    Delete(n,A,item);
    for(int i=0;i<n;i++,A++)cout<<*A<<"\n"[i==n-1];
}
int main(){
    int Aarray[5]={3,2,3,1,3},nArray=5,nList=5,item=3;
    node_iterator<int> Alist=new node<int>,cur=Alist;
    for(int i=0;i<nArray;i++)
        cur.Link(Aarray[i]);
    TEST(nArray,Aarray,item);
    TEST(nList,Alist,item);
}

```

- ```

template<class S,class T>void Delete(int& n,S A,const T& item)
{
 int m=0;
 S B=A;
 do{
 if(*A!=item){
 *B=*A;
 B++;
 m++;
 }
 }while(A++,--n);
 n=m;
}

```