

POWERSET

羅冠穎

2024/10/21

解題說明

依序瀏覽集合內的元素並分為選擇和不選擇兩種,將選擇元素記錄下來,瀏覽完所有元素後,開始輸出紀錄的元素

子集合數量為 2^N (N 為集合元素總數)

```
void subset(int* s,int n,int* store,int sv,int index){
    if(index==n){
        print_subset(store,sv);
        return;
    }

    subset(s,n,store,sv,index+1);

    subset(s,n,store,sv,index+1);

    store[sv]=0;
}
```

設計&實作

```
int main(){
    int N=0;//元素數量;
    cout<<"集合元素數量設置:";
    cin>>N;//決定集合大小
    int S[N]={0};;
    int store[N];//子集合元素暫存用
    cout<<"S=";
    for(int i=0;i<N;i+=1){
        cin>>S[i];
    }
    cout<<"以下為它的冪集合\n";
    subset(S,N,store,0,0);//(S集合, 暫存子集合, 索引值)
}
```

設計&實作

```
//找出子集合
void subset(int* s,int n,int* store,int sv,int index){//(S集合,S集合的元素數量,暫存子集合的元素,暫存索引值,索引值)
    if(index==n){//如果瀏覽完所有元素就開始打印
        print_subset(store,sv);
        return;
    }
    //不選擇當前元素,看下一個元素
    subset(s,n,store,sv,index+1);

    //選擇當前元素,之後看下一個元素
    store[sv++]=s[index];//將當前元素複製到子集合暫存
    subset(s,n,store,sv,index+1);

    //釋放當前元素
    store[sv]=0;
}
```

設計&實作

```
//打印子集合
void print_subset(int store[],int sv){
    cout<<"{";
    for(int x=0;x<sv;x+=1){
        cout<<store[x];
        if(x<sv-1){
            cout<<",";
        }
    }
    cout<<"}\n";
}
```

效能分析

時間複雜度:

$$T(P) = O(N * 2^N)$$

N為元素數量

空間複雜度:

$$S(P) = O(N)$$

N個元素

測試&過程

```
集合元素數量設置:3  
S=1 2 3  
以下為它的冪集合  
{}  
{3}  
{2}  
{2,3}  
{1}  
{1,3}  
{1,2}  
{1,2,3}  
  
-----  
Process exited after 4.725 seconds with return value 0  
請按任意鍵繼續 . . . |
```

驗證:

當 $\text{index} = N$ 時,進入遞迴處理第 $N+1$ 個元素遞迴產生兩條分支:

不選擇:進入下一層遞迴,處理下個元素

選擇:將當前元素加入暫存子集合,進入下一層遞迴,處理下個元素。以此類推直到瀏覽完所有元素。

遞迴不斷回溯,生成所有子集合,對於 M 個元素,輸出的結果應為 2^M 個子集合

心得&申論

心得:

作業寫完,我棒棒,希望成績也棒棒!!

申論:

這段程式碼實現了冪集合的生成，展示了遞迴算法的應用。

透過遞迴，程式能夠有效地列出所有子集合，並將結果打印出來。

這不僅有助於理解集合論的基本概念，還能增進對遞迴和算法設計的掌握，對學習計算機科學的學生非常好。