

類 科：資訊處理

科 目：資料結構

考試時間：2 小時

座號：_____

※注意：(一)禁止使用電子計算器。

(二)不必抄題，作答時請將試題題號及答案依照順序寫在試卷上，於本試題上作答者，不予計分。

- 一、給一個排序好的陣列 (Sorted Array) $A[\text{low}..\text{high}]$ ，當我們要搜尋一個元素 X 是否在此陣列 A 中，二元搜尋法 (Binary Search) 是檢查陣列的中間位置的元素 $A[\text{next}]$, $\text{next} = \lceil (\text{low} + \text{high}) / 2 \rceil$ ，和 X 做比較，並依比較結果作下列更新。

Case：

 $A[\text{next}] = X$: return $A[\text{next}] > X$: $\text{high} \leftarrow \text{next} - 1$ $A[\text{next}] < X$: $\text{low} \leftarrow \text{next} + 1$

重複上述步驟搜尋更新的陣列 $A[\text{low}..\text{high}]$ 直到找到 X 或確認 X 不是在此陣列 A 中。若我們設計一個新的搜尋法來修改二元搜尋法，每次都是以下列方式選取 $A[\text{next}]$ 。
 $\text{next} \leftarrow \text{low} + \lceil (\text{high} - \text{low}) * (X - A[\text{low}]) / (A[\text{high}] - A[\text{low}]) \rceil$

其他步驟都和二元搜尋相同。請回答下列問題：（每小題 5 分，共 15 分）

(一)新的搜尋法特色為何？請說明之。

(二)新的搜尋法在何種情形下，會比二元搜尋的搜尋速度為佳？請說明之。

(三)新的搜尋法，在最差的情況下，它的執行時間複雜度為多少？原因為何？假設陣列 A 中有 n 個元素。

- 二、 L 為一鏈結串列 (Linked List)，函數 $\text{Reverse}(L)$ 是要求把在原來 L 的每個節點 (Node) 的地址指標 (Pointer)，更改為指向它在鏈結串列 L 中的前面一個節點。請設計一個以疊代 (Iterative) 方式的程式來執行函數 $\text{Reverse}(L)$ 的功能，程式限制只能使用常數個 (constant) 額外空間 (External Memory)，可用程式語言 C、C++、Java 或 Pseudocode，寫出你的答案。請先說明你的作法，再寫出程式。（15 分）

- 三、若只能使用下列 6 種方式排序 (Sorting)：(a) Insertion Sort (b) Radix Sort (c) Merge Sort (d) Counting Sort (e) Heap Sort (f) Quick Sort。在下列各情形下，應選擇上述何種排序方法為最佳？請說明原因。（每小題 5 分，共 15 分）

(一)只要將全部資料中的前 20 名最大值排序好，並且主記憶體空間足夠。

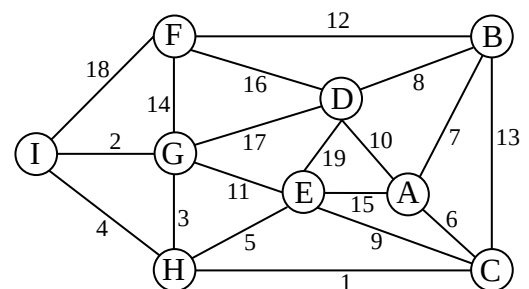
(二)只有少數資料在被已排序好的資料修改過，需要重排序，並且主記憶體空間足夠。

(三)資料無明顯特性，需要做第一次的排序，並且主記憶體空間足夠。

- 四、如右的權重圖 (weighted graph) 共有 9 個節點 (vertices) 19 條邊 (edges)，回答下列問題：

(一)請列出在運用 Kruskal's 演算法產生最小連結樹 (Minimum Spanning Tree) 中把邊納入最小連結樹的順序。（3 分）

(二)請列出運用 Prim's 演算法從 A 點開始產生最小連結樹，把邊納入最小連結樹的順序。（4 分）

(三)設計一個 $O(V)$ 的演算法，判定在新增加一個 (x,y) 的邊到原圖形後，是否要更新已經產生的最小連結樹。（8 分）

(請接背面)

類 科：資訊處理
科 目：資料結構

五、若處理的資料，其數值均不同且已知均為 1 到 100 之間的整數或小數。若 $K \leq X < K+1$ ，集合 L_x 代表數值在 $[K, K+1]$ 間全部資料， $1 \leq K \leq 99$, K 為整數，資料結構支援下列功能。

1. Insert(X)：增加 X ，若 X 不存在 L_x 中。
2. Delete(X)：移除 X ，若 X 存在 L_x 中。
3. List(X)：將 L_x 中的資料全部依序印出。

設計一資料結構滿足在最差情況的條件分析 (Worst Case Analysis)，每個功能的執行時間要求為：Insert(X) and Delete(X) 須在 $O(\log |L_x|)$ 時間內完成，List(X) 則須在 $O(|L_x|)$ 時間內完成。請說明設計的資料結構為何？並解釋其執行時間為何滿足需求？(15 分)

六、若 $G=(U,E)$ 為一權重圖 (weighted graph)，每條邊的權重均不為負數，則單源最短路徑問題 (Single Source Shortest Path Problem) 可以用著名的 Dijkstra 演算法求得，回答下列問題：(每小題 5 分，共 15 分)

(一)說明 Dijkstra 演算法的主要觀念。

(二)Dijkstra 演算法在最差情況下 (Worst Case Analysis)，下列三個功能 Insert、Delete、Decrease_Key 各自需要執行的次數，可用 Big-Oh 符號表示。

(三)若是要在 $O(|E| + |V| \log |V|)$ 最差情況分析下的時間內執行 Dijkstra 演算法，請問該選擇使用那種資料結構，並說明其原因。

七、下面二小題各有一段程式，其執行的時間是以執行 `sum++` 的次數計算，請用 Θ -notation 表示其執行時間，並說明其理由。(每小題 5 分，共 10 分)

(一)`sum=0`

```
for(i=0; i<2*n; i++)  
    for(j=0; j<i; j++)  
        sum++;
```

(二)`sum=0`

```
for(i=1; i<2*n; i++)  
    for(j=1; j<i*i; j++)  
        for(k=1; k<j; k++)  
            if(j%i==1)  
                sum++;
```