

資料結構與進階程式設計 (107-2)

作業二

作業設計：孔令傑
國立臺灣大學資訊管理學系

繳交作業時，請將第一題的答案以中文或英文作答後，以 PDF 檔上傳到 NTU COOL；**不接受紙本繳交**；第二、三題請至 PDOGS (<http://pdogs.ntu.im/judge/>) 上傳一份 C++ 原始碼 (以複製貼上原始碼的方式上傳)。這份作業的截止時間是 **2019 年 3 月 25 日凌晨 1:00**。在你開始前，請閱讀課本的第 14 和 16 章¹。不接受遲交。

由於本課程修課同學眾多，而助教人力有限，為了避免過度壓榨助教，我們可能會對手寫題進行隨機批改，亦即助教會隨機選擇一份作業的若干題進行批改和給分。助教會對全班同學都批改同樣的題目。若有批改之題目的總分為 x 而一位同學得到 y 分，且手寫題總分為 z 分，則其在該份作業會得到 $z(\frac{y}{x})$ 分。公佈的參考解答中會包含所有題目 (包含沒被批改的)。

第一題

(40 分，每小題 10 分) 請考慮下面的程式：

```
#include <iostream>
using namespace std;
int getArrayElement(int array[], int len, int index);
int main() {
    int array[10] = {0};
    for(int i = 0; i < 10; i++)
        cin >> array[i];
    int index = 0;
    cin >> index;
    int result = getArrayElement(array, 10, index);
    cout << if(result == -1)
        cout << "Bad index!\n";
    else
        cout << result << endl;
    return 0;
}
int getArrayElement(int array[], int len, int index) {
    if(0 <= index && index < len)
        return array[index];
    else
        return -1;
}
```

¹課本是 Deitel and Deitel 著的 *C++ How to Program: Late Objects Version* 第七版。

`getArrayElement` 這個函數做的事情很簡單：給定一個陣列、它的長度，以及一個 `array index`，如果該 `index` 合法，就回傳存在該 `index` 的值，否則就回傳 `-1`。

- (a) 這個程式有至少四個缺點：「10」這個數字被寫死在程式碼的複數個地方、`getArrayElement` 的 `function header` 不保證 `array` 不會在函數裡被改掉、不能保證 `main function` 的開發者會檢查回傳值，以及如果陣列裡面可能本來就會存 `-1` 那就很困擾。請修改 `getArrayElement`，使用任意方法解決前兩個問題，並使用 `exception handling` 的方法解決後兩個缺點；當然你不一定要回傳 `-1` 了。接著請修改 `main function` 以做到題目中的程式做到的事。
- (b) 接續妳在前一小題寫的程式，請接著用 `template` 改寫 `getArrayElement`，讓這個函數可以接受 `double array`。
- (c) 承上題，令 n 為陣列 `array` 的長度（即傳入的 `len`），請修改 `getArrayElement`，讓它接收到的 `index` 介於 -1 到 $-(n-1)$ 時，會回傳倒著數回來的第 $-\text{index}$ 個元素，例如收到 `-1` 則回傳倒數第一個（第 $n-1$ 個）、收到 `-2` 則回傳倒數第二個（第 $n-2$ 個），依此類推。其他情況請仍以 `exception` 做處理。
- (d) 承上題，請問現在的 `getArrayElement` 可以正確地處理物件陣列嗎？就是說，如果傳入的 `array` 不是 `int`、`double` 這類基本型態，而是 `MyVector`、`Warrior` 這類的 `class` 型態，它還能正確地運作嗎？還是你需要多寫些什麼？

第二題

（20 分）承作業一第三題，某港的地形呈現如圖 1。如圖所示，船舶在「進港」時會行駛兩段路程，第一段為從外航站到迴船池的路程，第二段為從迴船池到各停靠碼頭的路程；在「出港」時也會行駛同樣的兩段路程，順序反過來；在「移泊」時只會行駛一段路程，即港內各停靠碼頭間的移動。因此，「進港」或「出港」的船舶作業時間會改成記錄兩個數字， p_1 、 p_2 分別為第一段與第二段路程所耗費的作業時間（單位：分鐘，為 1-200 間的整數），加總即為總作業時間（ $p = p_1 + p_2$ ）。請注意出港的第一段是碼頭到迴船池，而進港的第一段是外航站到迴船池。

顯然地，水道衝突應該只會發生在迴船池和外航站之間，也就是說作業一第三題的「互不衝突」規則是嚴謹得過頭了。舉例來說，如果有條進港船從外航站行駛進港並在 10:20 抵達迴船池，則下一條出港船舶便可以在 10:20 開始從迴船池前往外航站行駛出港。

在本題，給定的船舶與工作資訊、兩種演算法運作原則與相關假設條件皆如同作業一第三題所述，差別在於以下兩處。首先，「工作資訊」檔案中對於船舶作業時間的記載方面，出港與進港船會記錄兩個數字，分別是該工作的第一段與第二段路程作業時間，而移泊船則依然只有一個數字。其次演算法的「互不衝突」規則修改為一個水道一次只能通過一條船舶，而且每一條船舶都要「完整地駛離水道」才能換下一條船舶開。

因此，對於船舶排班只須注意「避免水道衝突」，即各船舶實際行駛於水道的時間不可以重疊。舉例來說，若進港船舶 `AFECXS` 的可開始時間是 10:00、第一段路程作業時間是 20 分鐘、第二段路程作業時間是 30 分鐘、從 1 港口進出、港內停靠於編號 12 碼頭，則該條船舶的工作資訊將被記載如下：

<code>AFECXS,10:00,20,30,I,1,12</code>
--

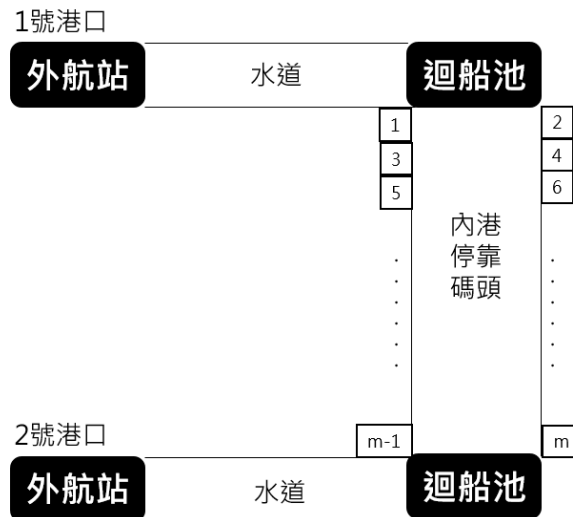


圖 1: 港口示意圖

可以看到原本只用一個數字記載的作業時間被拆成兩個數字（20 與 30），中間用一個逗點隔開。這表示船舶 AFECXS 預計 10:00 從外航站出發行駛進港並在 10:20 抵達迴船池。若出港船舶 TRUCXS 的可開始時間是 10:10、第一段路程作業時間是 10 分鐘、第二段路程作業時間是 5 分鐘、從 1 港口進出、港內停靠於編號 15 碼頭，則該條船舶工作資訊記載如下：

TRUCXS ,10:10,10,5,0,1,15

假設我們將該進港船舶 AFECXS 排定於 10:00 到 10:20 佔用水道，則出港船舶 TRUCXS 將可以於 10:20 從迴船池出發行駛出港並在 10:25 抵達外航站，亦即它也可以在它的可出發時間 10:10 就離開碼頭。在本題的規則下，上述這兩條船舶可以無縫接軌地使用水道，各自準時在其可出發時間出發而沒有延遲時間，不用像作業一第三題那樣讓出港船等到 10:50 才出發。

本題在給定船舶資訊與每日工作資訊的情況下，我們需要撰寫排班程式，分別來實踐作業一第三題指定的兩種排班演算法 Earliest Ready Time First (ERT) 和 Shortest Processing Time First (SPT)(依然使用總作業時間來排序)，並計算出在兩種演算法下各自所有工作的延遲時間總和來作輸出。

輸入輸出格式

系統會提供一共 10 組測試資料，每組測試資料裝在一個檔案裡。本題的輸入輸出格式和作業一第三題一模一樣，只差在工作資訊檔案的格式稍有不同。在工作資訊檔案中總共會有 n 行，每一行代表每件工作的相關資訊，若為進港或出港船則依序為船舶編號、可出發時間、第一段路程作業時間、第二段路程作業時間、類型、港口、起點碼頭、終點碼頭，若為移泊船則依序為船舶編號、可出發時間、作業時間、類型、港口、起點碼頭、終點碼頭。

舉例來說，若 ShipSample.txt 為船舶資訊檔案，其內容為

AFECXS ,25500 ,TAIWAN ,LING KUNG ,Y
 AFEUTI ,20000 ,JAPAN ,HIRONA OISHI ,N
 TRUCXS ,13700 ,THAILAND ,SAWADIKA ,Y

```
THUEDC,18600,TAIWAN,YENYENER,N
KCUHIC,78600,THAILAND,KAPUNKA,Y
ITCUOW,12000,TAIWAN,JASON WANG,Y
ZCHCBE,88888,JAPAN,KANAKO,N
```

而 WorkSample.txt 為工作資訊檔案，其內容為

```
AFECXS,10:00,30,20,I,1,12
AFEUTI,10:40,12,T,106,120
TRUCXS,10:10,10,5,0,1,15
THUEDC,10:10,10,10,0,1,35
KCUHIC,10:20,25,15,I,1,150
ZCHCBE,13:00,10,30,0,2,35
```

則若輸入是

```
200
ShipSample.txt
WorkSample.txt
```

則輸出應該是

```
50,80
```

你上傳的原始碼裡應該包含什麼

你的.cpp 原始碼檔案裡面應該包含讀取測試資料、做運算，以及輸出答案的 C++ 程式碼。當然，你應該寫適當的註解。針對這個題目，你**可以**使用任何方法。

評分原則

這一題的分數都根據程式運算的正確性給分。PDOGS 會編譯並執行你的程式、輸入測試資料，並檢查輸出的答案的正確性。一筆測試資料佔 2 分。

第三題

(40 分) 本題的情境與作業一第三題的一樣，但現在我們不做船舶排班演算法的規劃，而是做船舶作業時間的預測。在實務上，一條船舶要花多少時間才能完成其進港、出港、移泊作業，當然是做完才會知道，因此排程時是根據給定的船舶資訊與工作資訊，然而我們也可以利用某些預測方法去預測出預期的作業時間，再根據預期的作業時間來做排程規劃。

在本題中，我們將利用如下高度簡化的預測模型來預測船舶的作業時間。針對進港船舶的作業時間，預測模型為

$$y = a_0 + a_1 \times MovingDST + a_2 \times ShipWeight + a_3 \times Danger + a_4 \times Night。$$

出港船舶作業時間的預測模型為

$$y = b_0 + b_1 \times MovingDST + b_2 \times ShipWeight + b_3 \times Danger + b_4 \times Night。$$

移泊船舶作業時間的預測模型則為

$$y = c_0 + c_1 \times MovingDST + c_2 \times ShipWeight + c_3 \times Danger + c_4 \times Night。$$

上述作業時間預測模型的變數定義如下：

- *MovingDST*：即船舶行駛的移動距離，單位為公里。已知迴船池和外航站之間的距離為 t_1 ，迴船池和碼頭之間的距離則如圖 2 所示，垂直方向行駛的距離每段（迴船池與碼頭 1、迴船池與碼頭 2、碼頭 i 與碼頭 $i + 2$ 、碼頭 $m - 1$ 與迴船池、碼頭 m 與迴船池）皆為 t_2 ，水平方向行駛的距離每段（碼頭 i 與碼頭 $i + 1$ ）則皆為 t_3 。舉例來說，一條從一港口進港要抵達碼頭 6 的進港船的航行距離為 $t_1 + 3t_2 + t_3$ 、一條從碼頭 2 要移泊到碼頭 5 的移泊船的航行距離為 $2t_2 + t_3$ 。

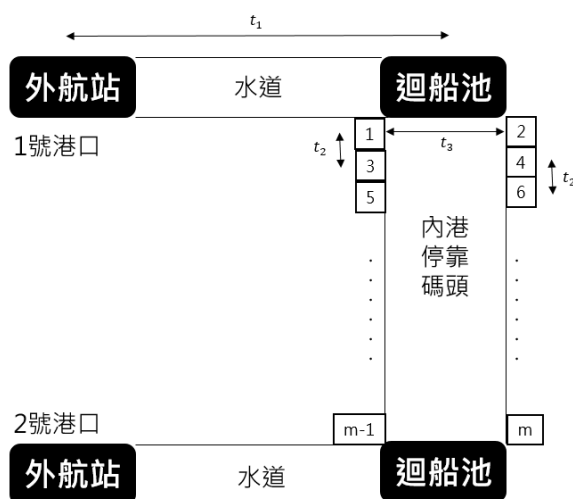


圖 2: 港口與距離資訊示意圖

- *ShipWeight*：即船舶的重量噸位，單位為噸。
- *Danger*：即船舶「是否具有危險性」，透過船舶資訊中「是否裝載危險貨物」來判斷，是則為 1，否則為 0。
- *Night*：即船舶是否在夜間出發，其夜間定義在於工作資訊中船舶的可出發時間是否在晚上六點到清晨六點之間（包含晚上六點和清晨六點），是則為 1，否則為 0。

由於預測模型的係數可能非整數，因此按照公式預測出 y 值之後，請將之無條件捨去到整數位做為預測的作業時間。

輸入輸出格式

系統會提供一共 15 組測試資料，每組測試資料裝在一個檔案裡。在每個檔案中會有若干行，第一行是給定的模型參數與係數，共有 19 個數字，依序為碼頭數 m 、進港預測模型的係數 a_0 到 a_4 出港預測模

型的係數 b_0 到 b_4 移泊預測模型的係數 c_0 到 c_4 以及距離長度 t_1 、 t_2 、 t_3 ，任兩個數字之間用一個逗點隔開。第二行會有一個檔案路徑的字串，表示船舶資訊檔案的存放位置。第三行會有一個檔案路徑的字串，表示工作資訊檔案的存放位置。十五個預測模型的參數都是介於 -1000 到 1000 之間的實數（可能非整數）， $m \in \{2, 4, \dots, 200\}$ 、三個距離長度則為介於 1 到 1000 之間的整數。

在本題中，船舶資訊檔案內容記載形式如同作業一第三題，工作資訊檔案內容記載形式也大同小異，差別之處僅有兩處。首先，工作資訊中不包含船舶作業時間；其次，船舶編號以外的記載資訊可能會有缺值，若有缺值時則該值將會被空字串取代。舉例來說，

```
AFEUTI,19:40,T,,120
TRUCXS,,0,1,15
KCUHIC,21:20,I,1,
```

表示 AFEUTI 的進港港口資訊缺失、TRUCXS 的可出發時間缺失，而 KCUHIC 的目標碼頭資訊缺失。若一個工作有任何一個缺失值，我們就放棄預測其作業時間。

在依序讀入船舶資訊與工作資訊後，請分兩列輸出運算結果。在第一列，請依序輸出每個工作的預測作業時間（按照工作資訊檔案中逐行的順序），若遇有缺值者則輸出 -1。換言之，若工作資訊檔案總計有 n 個工作，則第一列就會有共輸出 n 個數字。任兩個數字之間用一個逗點隔開，當然最後一個數字後面沒有逗點。在第二列請輸出總共有幾個工作有缺值。

舉例來說，若 AnotherAnotherShipSample.txt 為船舶資訊檔案，其內容為

```
AFECXS,25500,TAIWAN,LING KUNG,Y
AFEUTI,20000,JAPAN,HIRONA OISHI,N
TRUCXS,13700,THAILAND,SAWADIKA,Y
THUEDC,18600,TAIWAN,YENYENER,N
KCUHIC,78600,THAILAND,KAPUNKA,Y
ITCUOW,12000,TAIWAN,JASON WANG,Y
ZCHCBE,88888,JAPAN,KANAKO,N
```

而 AnotherAnotherWorkSample.txt 為工作資訊檔案，其內容為

```
AFECXS,10:00,I,1,12
AFEUTI,19:40,T,,120
TRUCXS,,0,1,15
KCUHIC,21:20,I,1,150
ZCHCBE,00:00,0,2,198
```

則如果輸入是

```
200,5,0.0225,0.007,5,5,3,0.015,0.007,4,5,4,0.02,0.005,7,4,7,1,9
AnotherAnotherShipSample.txt
AnotherAnotherWorkSample.txt
```

則最後輸出應該是

```
188,-1,-1,567,630
2
```

你上傳的原始碼裡應該包含什麼

你的.cpp 原始碼檔案裡面應該包含讀取測試資料、做運算，以及輸出答案的 C++ 程式碼。當然，你應該寫適當的註解。針對這個題目，你**不可以**使用上課沒有教過的方法。

請在讀進給定的資訊時，生成船舶與工作物件，此時工作物件中的作業時間應該是空的。接著請寫一個函數，其參數為一個工作物件，該函數會幫傳入的工作物件填入預測出的作業時間。你應該要有三個工作類別（進港、出港、移泊），而剛剛的函數參數則為這三個工作類別的基底類別。若給定的工作資訊有缺漏（例如沒有記錄進港船的目標碼頭），此時該函數應該 throw exception，並由 catch exception 的程式碼做有缺值資料筆數的統計。

評分原則

- 這一題的其中 30 分會根據程式運算的正確性給分。PDOGS 會直譯並執行你的程式、輸入測試資料，並檢查輸出的答案的正確性。一筆測試資料佔 2 分。
- 這一題的其中 10 分會根據你所寫的程式的品質來給分。助教會打開你的程式碼並檢閱你的程式，而評分重點放在你使用的繼承、多型與 exception handling。