國立屏東大學 資訊工程系 物件導向程式設計

Turnin 作業 5

- Turnin Code: cpp.hw5
- Due Date: 4/20 Monday 23:59:00 (midnight) **Hard Deadline**

繳交方式說明

本次作業繳交將以資料夾的形式繳交，需要為每一題建立一個資料夾（資料夾名稱為該題題目前方之代號，第一題為p1第二題為p2餘以此類推）。

繳交說明可參考Turnin 作業 4】中 p6 到 p10

任何未依照正確繳交格式的檔案將以 0 分計。



本文使用「 」及「 」代表「空白字元」與Enter 換行字元」，並且將使用者輸入的部份使用灰階方式顯示。另外，題目的執行結果中，如果出現「(」、「)」、「:」、「;」、「.」與「,」等符號，皆為英文半形！

---

p1 兩數交換

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "swap.h"
#include <iostream>
using namespace std;
int main()
{
    int a, b;
    cout << "Please input two integers(a b): ";
    cin >> a >> b;
    cout << "Before swapping: a = " << a << " and b = " << b << endl;
    swap(&a, &b);
    cout << "After swapping:  a = " << a << " and b = " << b << endl;
```

```
    return 0;
}
```

請完成名為 **swap.cpp** 與 **swap.h** 的 C++ 語言程式，其中分別包含 swap 函式的 **Implementation** 與其 **Prototype 宣告**。

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp swap.o
    g++ main.cpp swap.o
swap.o: swap.cpp swap.h
    g++ -c swap.cpp
clean:
    rm -f a.out *.c~ *.o*
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p1]▲./a.out↵
Please▲input▲two▲integers(a▲b):▲3▲7↵
Before▲swapping:▲a▲=▲3▲and▲b▲=▲7↵
After▲swapping:▲▲a▲=▲7▲and▲b▲=▲3↵
[3:23▲user@ws▲p1]▲./a.out↵
Please▲input▲two▲integers(a▲b):▲-13▲216↵
Before▲swapping:▲a▲=▲-13▲and▲b▲=▲216↵
After▲swapping:▲▲a▲=▲216▲and▲b▲=▲-13↵
[3:23▲user@ws▲p1]▲
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
    - swap.h
    - swap.cpp

p2 簡易計算機

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "calculator.h"
#include <iostream>
```

```
using namespace std;
int main()
{
    double a = 0, b = 0, ans = 0;
    char op = '\0'; // operator

    while (true)
    {
        cout << ">> ";
        cin >> a;
        if (a == -127)
        {
            cout << "Calculator terminated successfully." << endl;
            return 0;
        }
        cin >> op >> b;

        switch (op)
        {
            case '+':
            {
                add(a, b, ans);
                break;
            }
            case '-':
            {
                sub(a, b, ans);
                break;
            }
            case '*':
            {
                mult(a, b, ans);
                break;
            }
            case '/':
            {
                if (b == 0)
                {
                    cout << "Warning! Division by zero!" << endl;
                    continue;
                }
                else
                {
                    divi(a, b, ans);
                }
                break;
            }
            default:
                cout << "Unreadable operator!" << endl;
```

```
                break;
            }
            cout << ans << endl;
        }
        return 0;
}
```

請完成名為 **calculator.cpp** 與 **calculator.h** 的 C++ 語言程式，其中分別包含 add、sub、mult 與 divi 函式的 **Implementation** 與其 **Prototype 宣告**。

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp calculator.o
    c++ main.cpp calculator.o
calculator.o: calculator.cpp calculator.h
    c++ -c calculator.cpp
clean:
    rm -f a.out *.c~ *.o*
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p2]▲./a.out↵
>>▲4+5↵
9↵
>>▲273.15-100.15↵
173↵
>>▲216/6↵
36↵
>>▲45*4↵
180↵
>>▲90/12↵
7.5↵
>>▲2/3↵
0.666667↵
>>▲-127↵
Calculator▲terminated▲successfully.↵
[3:23▲user@ws▲p2]▲
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
  - calculator.h
  - calculator.cpp

p3 手機號碼格式檢查

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "phone_number.h"
#include <iostream>
using namespace std;

int main()
{
    string phone_number;
    cout << "Please enter your phone number: ";
    getline(cin, phone_number);
    if (!is_correct_format(phone_number))
    {
        cout << "[" << phone_number << "] is invaild phone number format!"
<< endl;
        return -1;
    }
    cout << "Your phone number is ["
         << formatted(phone_number) << "], which is valid." << endl;
    return 0;
}
```



第十行 is_correct_fromat 已更正為 is_correct_format

使用者將輸入一個行動電話號碼，請判斷該門號是否有效：

- 為 10 個字元
- 前兩個字元為 "09"
- 每個字元都是 0～9 的數字

最後，若該號碼為有效的格式，請將號碼依照xxxx-xxxxxx格式輸出。

請完成名為 **phone_number.cpp** 與 **phone_number.h** 的 C++ 語言程式，其中分別包含 is_correct_fromat 與 formatted 函式的 **Implementation** 與其 **Prototype 宣告**

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp phone_number.o
    g++ main.cpp phone_number.o
phone_number.o: phone_number.cpp phone_number.h
    g++ -c phone_number.cpp
```

```
clean:
    rm -f *.o *~ *.*~ a.out
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p3]▲./a.out↵
Please▲enter▲your▲phone▲number:▲0912345678↵
Your▲phone▲number▲is▲[0912-345678],▲which▲is▲valid.↵
[3:23▲user@ws▲p3]▲./a.out↵
Please▲enter▲your▲phone▲number:▲9012345678↵
[9012345678]▲is▲invaild▲phone▲number▲format!↵
[3:23▲user@ws▲p3]▲./a.out↵
Please▲enter▲your▲phone▲number:▲90123456789↵
[90123456789]▲is▲invaild▲phone▲number▲format!↵
[3:23▲user@ws▲p3]▲./a.out↵
Please▲enter▲your▲phone▲number:▲091234567S↵
[091234567S]▲is▲invaild▲phone▲number▲format!↵
[3:23▲user@ws▲p3]▲
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
    - phone_number.h
    - phone_number.cpp

p4 字串修剪

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "trim.h"
#include <iostream>
using namespace std;

int main()
{
    string str;
    cout << "Please enter a string: ";
    getline(cin, str);
    trim(str);
    cout << "Trimmed string is [" << str << "]" << endl;
    return 0;
```

```
}
```

使用者將輸入一個字串，請將該字串前後的空白移除（即移除字串中第一個不為空白的字元前的所有空白，以及最後一個不為空白的字元後的所有空白）。請完成名為 **trim.cpp** 與 **trim.h** 的 C++ 語言程式，其中分別包含 trim 函式的 **Implementation** 與其 **Prototype 宣告**。

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp trim.o
    g++ main.cpp trim.o
trim.o: trim.cpp trim.h
    g++ -c trim.cpp
clean:
    rm -f *.o *~ *.*~ a.out
```

本題將提供一些字串供同學們測試，如下所示：

```
測試檔 1：[        I hear Jerusalem bells a-ringin' Roman cavalry choirs are
singin' Be my mirror, my sword and shield My missionaries in a foreign field
]

測試檔 2：[     snake on the beach  -^~~~~~~~~~    ]

測試檔 3：[      ~ - Come on, get on up We're wild and we can't be tamed And
we're turnin' the floor into A zoo, ooh, ooh      !   ]
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p4]▲./a.out↵
Please▲enter▲a▲string:▲▲▲▲▲▲▲▲▲I▲hear▲Jerusalem▲bells▲a-
ringin'▲Roman▲cavalry▲choirs▲are▲singin'▲Be▲my▲mirror,▲my▲sword▲and▲shiel
d▲My▲missionaries▲in▲a▲foreign▲field▲▲▲▲▲↵
Trimmed▲string▲is▲[I▲hear▲Jerusalem▲bells▲a-
ringin'▲Roman▲cavalry▲choirs▲are▲singin'▲Be▲my▲mirror,▲my▲sword▲and▲shield▲My▲missionaries▲
in▲a▲foreign▲field]↵
[3:23▲user@ws▲p4]▲./a.out↵
Please▲enter▲a▲string:▲▲▲▲▲▲snake▲on▲the▲beach▲▲-^~~~~~~~~~▲▲▲▲↵
Trimmed▲string▲is▲[snake▲on▲the▲beach▲▲-^~~~~~~~~~]↵
[3:23▲user@ws▲p4]▲./a.out↵
Please▲enter▲a▲string:▲▲▲▲▲▲▲~▲-
Come▲on,▲get▲on▲up▲We're▲wild▲and▲we▲can't▲be▲tamed▲And▲we're▲turnin'▲the
▲floor▲into▲A▲zoo,▲ooh,▲ooh▲▲▲▲▲▲!▲▲▲↵
Trimmed▲string▲is▲[~▲-
Come▲on,▲get▲on▲up▲We're▲wild▲and▲we▲can't▲be▲tamed▲And▲we're▲turnin'▲the▲floor▲into▲A▲zo
```

o,▲ooh,▲ooh▲▲▲▲▲▲!]↵
[3:23▲user@ws▲p4]▲



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
  - trim.h
  - trim.cpp

p5 檢查回文

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "palindrome.h"
#include <iostream>
using namespace std;
int main()
{
    string str;
    cout << "Please enter a string: ";
    getline(cin, str);
    cout << "[" << str << "] ";
    if (is_palindrome(str))
    {
        cout << "is a palindrome string.";
    }
    else
    {
        cout << "is not a palindrome string.";
    }
    cout << endl;
    return 0;
}
```

所謂「回文」指的是無論從左往右讀，還是從右往左讀，拼法都一模一樣的字串。使用者將輸入一個字串，請去除所有非字母的字元（例如：數字、空格、標點與其他特殊字元）後再判斷該字串是否為回文字串。請完成名為 **palindrome.cpp** 與 **palindrome.h** 的 C++ 語言程式，其中分別包含 is_palindrome 函式的 **Implementation** 與其 **Prototype 宣告**

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp palindrome.o
```

```
    g++ main.cpp palindrome.o
palindrome.o: palindrome.cpp palindrome.h
    g++ -c palindrome.cpp
clean:
    rm -f *.o *~ *.*~ a.out
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p5]▲./a.out↵
Please▲enter▲a▲string:▲raceCaR↵
[raceCaR]▲is▲a▲palindrome▲string.↵
[3:23▲user@ws▲p5]▲./a.out↵
Please▲enter▲a▲string:▲acecar↵
[acecar]▲is▲not▲a▲palindrome▲string.↵
[3:23▲user@ws▲p5]▲./a.out↵
Please▲enter▲a▲string:▲Was▲it▲a▲cat▲I▲saw?↵
[Was▲it▲a▲cat▲I▲saw?]▲is▲a▲palindrome▲string.↵
[3:23▲user@ws▲p5]▲./a.out↵
Please▲enter▲a▲string:▲Was▲it▲a▲can▲I▲saw?↵
[Was▲it▲a▲can▲I▲saw?]▲is▲not▲a▲palindrome▲string.↵
[3:23▲user@ws▲p5]▲./a.out↵
Please▲enter▲a▲string:▲nurses▲run!!!↵
[nurses▲run!!!]▲is▲a▲palindrome▲string.↵
[3:23▲user@ws▲p5]▲./a.out↵
Please▲enter▲a▲string:▲A▲Santa▲at▲NASA↵
[A▲Santa▲at▲NASA]▲is▲a▲palindrome▲string.↵
[3:23▲user@ws▲p5]▲./a.out↵
Please▲enter▲a▲string:▲borrow▲10000▲or▲rob▲1000000000↵
[borrow▲10000▲or▲rob▲1000000000]▲is▲a▲palindrome▲string.↵
[3:23▲user@ws▲p5]▲
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
  - palindrome.h
  - palindrome.cpp

p6 字串與記憶體位址

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "str_addr.h"
#define SIZE 1024
int main()
{
    char str[SIZE];
    char *ptr = str; // pointer
    int length = 0;
    cout << "Please input a string: ";
    get_string(ptr, SIZE);

    for (int i = 0; str[i] != '\0'; i++)
    {
        print_substr_addr(ptr+i);
    }
    return 0;
}
```

本題接受使用者輸入的一個字串，並使用 ptr 指標變數指向 str 字元陣列，並輸出當前指標所指向的記憶體位址與字串。每完成一次輸出，指標將向後移動至下一個字元，輸出當前指標所指向的記憶體位址，並且輸出自當前指標到字串的截止符號 '\0' 前的子字串，重複此過程直到 ptr 指向字串的終點。最後一次輸出時將呈現完整的字串內容。

請完成名為 **str_addr.cpp** 與 **str_addr.h** 的 C++ 語言程式，其中分別包含 get_string 與 print_substr_addr 函式的 **Implementation** 與其 **Prototype 宣告**

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp str_addr.o
    g++ main.cpp str_addr.o
str_addr.o: str_addr.cpp str_addr.h
    g++ -c str_addr.cpp
clean:
    rm -f *.o *~ *.*~ a.out
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p6]▲./a.out↵
Please▲input▲a▲string:▲I▲am▲the▲greatest▲programmer▲on▲Earth!↵
[0xffffcb587ab8][I▲am▲the▲greatest▲programmer▲on▲Earth!]↵
[0xffffcb587ab9][▲am▲the▲greatest▲programmer▲on▲Earth!]↵
[0xffffcb587aba][am▲the▲greatest▲programmer▲on▲Earth!]↵
[0xffffcb587abb][m▲the▲greatest▲programmer▲on▲Earth!]↵
[0xffffcb587abc][▲the▲greatest▲programmer▲on▲Earth!]↵
[0xffffcb587abd][the▲greatest▲programmer▲on▲Earth!]↵
[0xffffcb587abe][he▲greatest▲programmer▲on▲Earth!]↵
[0xffffcb587abf][e▲greatest▲programmer▲on▲Earth!]↵
[0xffffcb587ac0][▲greatest▲programmer▲on▲Earth!]↵
```

```
[0xffffcb587ac1][greatest▲programmer▲on▲Earth!]↵
[0xffffcb587ac2][reatest▲programmer▲on▲Earth!]↵
[0xffffcb587ac3][eatest▲programmer▲on▲Earth!]↵
[0xffffcb587ac4][atest▲programmer▲on▲Earth!]↵
[0xffffcb587ac5][test▲programmer▲on▲Earth!]↵
[0xffffcb587ac6][est▲programmer▲on▲Earth!]↵
[0xffffcb587ac7][st▲programmer▲on▲Earth!]↵
[0xffffcb587ac8][t▲programmer▲on▲Earth!]↵
[0xffffcb587ac9][▲programmer▲on▲Earth!]↵
[0xffffcb587aca][programmer▲on▲Earth!]↵
[0xffffcb587acb][rogrammer▲on▲Earth!]↵
[0xffffcb587acc][ogrammer▲on▲Earth!]↵
[0xffffcb587acd][grammer▲on▲Earth!]↵
[0xffffcb587ace][rammer▲on▲Earth!]↵
[0xffffcb587acf][ammer▲on▲Earth!]↵
[0xffffcb587ad0][mmer▲on▲Earth!]↵
[0xffffcb587ad1][mer▲on▲Earth!]↵
[0xffffcb587ad2][er▲on▲Earth!]↵
[0xffffcb587ad3][r▲on▲Earth!]↵
[0xffffcb587ad4][▲on▲Earth!]↵
[0xffffcb587ad5][on▲Earth!]↵
[0xffffcb587ad6][n▲Earth!]↵
[0xffffcb587ad7][▲Earth!]↵
[0xffffcb587ad8][Earth!]↵
[0xffffcb587ad9][arth!]↵
[0xffffcb587ada][rth!]↵
[0xffffcb587adb][th!]↵
[0xffffcb587adc][h!]↵
[0xffffcb587add][!]↵
[3:23▲user@ws▲p6]▲
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題將採人工批改。
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
  - str_addr.h
  - str_addr.cpp

p7 解析字串並計算學期成績

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "term_score.h"
#define SIZE 1024
int main()
{
    char str[SIZE];
    char *ptr = str; // pointer
    int length = 0;
    cout << "Please input a score information: ";
    get_string(ptr, SIZE);
    data_processing(ptr);
}
```

本題接受使用者輸入的一個字串，字串的格式為「學號:姓名:期中考成績:期末考成績:平時成績　」。請解析使用者輸入的字串並計算學期成績（四捨五入到小數點第一位）：

- 期中考成績 35%
- 期末考成績 35%
- 平時成績 30%

請完成名為 **term_score.cpp** 與 **term_score.h** 的 C++ 語言程式，其中分別包含 get_string 與 data_processing 函式的 **Implementation** 與其 **Prototype 宣告**

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp term_score.o
    g++ main.cpp term_score.o
term_score.o: term_score.cpp term_score.h
    g++ -c term_score.cpp
clean:
    rm -f *.o *~ *.*~ a.out
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p7]▲./a.out↵
Please▲input▲a▲score▲information:▲cbb113000:Jun▲Wu:97.4:94:82.4↵
Jun▲Wu(cbb113000):▲91.7↵
[3:23▲user@ws▲p7]▲./a.out↵
Please▲input▲a▲score▲information:▲SuspiciousSoup003:Usagi:63:72:91.7↵
Usagi(SuspiciousSoup003):▲74.8↵
[3:23▲user@ws▲p7]▲
```



同學們在分割字串的時候除了自己手動處理遇到每一個冒號時將前面的字串全部取出來，還可以使用定義在 <cstring> 中的 strtok() 函式，請參考下列程式碼：



```
char str[1024] = "Snape was very angry! \"You dare use my own spells against me, Potter?\" said Snape.";
char * ptr = str;

strtok(ptr, " ,.\":!?"); // 若遇到半形的「空白」、「逗點」、「句點」、「雙引號」、「冒號」、「驚嘆號」、「問號」就分割（tokenize）字串
// 上述的符號我們稱為分隔符號（delimiter）
while (ptr != NULL) { // 若字串分割後還有內容則繼續迴圈（繼續分割）
    cout << ptr << endl;
    ptr = strtok(NULL, " ,.\":!?");
  // 填入 NULL 表示根據上次切割的地方開始繼續往下找分隔符號，並讓 ptr 指標指向下一個分隔符後面的字串，
  // 如果傳回 NULL 表示字串已經全部分割完成，迴圈不符合繼續執行的條件，結束迴圈
}
// 請參考 https://cplusplus.com/reference/cstring/strtok/?kw=strtok
```

以下為該程式的執行結果：

```
Snape
was
very
angry
You
dare
use
my
own
spells
against
me
Potter
said
Snape
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題將採人工批改。
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
  - term_score.h
  - term_score.cpp

p8 我所知道的禽類--徐志摩沒說過（寬字串）

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "bird.h"

int main()
{
    ios::sync_with_stdio(false);
    wcin.imbue(locale("C.UTF-8"));
    wcout.imbue(locale("C.UTF-8"));
    wstring database[][2] = {
        {L"亞洲輝椋鳥", L"城鎮和城市中大量分佈，牠們棲息於廢棄的建築物和樹木中，經常成群活動，被認為是最吵鬧的鳥類之一"},
        {L"紅鳩", L"全身覆蓋著亮眼的酒紅色羽毛，頭部至頸部則呈鼠灰色，後頸有一道明顯的黑色頸環，就像是脖子掛著頭戴式耳機一樣"},
        {L"金背鳩", L"因爲翅膀覆羽具備顯眼的紅褐色或金棕色邊緣，在陽光下閃爍金色光芒而得名"},
        {L"珠頸斑鳩", L"頸部後方有一塊明顯的黑色白點斑塊，像是戴了珍珠項鍊，是都市裡最常見的「咕顧估」"},
        {L"灰斑鳩", L"全身羽毛呈灰淺褐色，頸後有一條細長的黑環，就像是脖子掛著頭戴式耳機一樣"},
        {L"鴿", L"英文俗名□pigeons□和「dove□在中文可以譯成鴿或鳩，基本上，較小的叫dove□較大的叫pigeon"},
        {L"鳩", L"英文俗名□pigeons□和「dove□在中文可以譯成鴿或鳩，基本上，較小的叫dove□較大的叫pigeon"},
        {L"愛蓮野鴿", L"進擊的巨人主角，一言不合就地鳴"},
        {L"南南東", L"竈門炭治郎的鎹鴉（天王寺松右衛門）常用的任務指示方位"},
        {L"黑冠麻鷺", L"俗稱「大笨鳥」，在草地上常維持縮頭不動的姿勢，受驚嚇時會伸長脖子偽裝成樹枝"},
        {L"和尚鸚鵡", L"前額與胸部呈淺灰色，像披了一件僧侶的斗篷，故得名「和尚」"},
        {L"金剛鸚鵡", L"體型巨大的長尾鸚鵡，羽色極其鮮豔，智商高且壽命極長，能輕易咬碎堅硬的果殼"},
        {L"玄鳳鸚鵡", L"又稱雞尾鸚鵡，最明顯的特徵是頭頂的冠羽和臉頰兩側的橘色圓斑，看起來就像香蕉形狀的皮卡丘"},
        {L"凱克鸚鵡", L"鸚鵡界的「跳跳虎」，喜歡跳躍式前進，腹部羽毛雪白，性格頑皮愛玩"},
        {L"非洲灰鸚鵡", L"公認智商最高的鳥類之一，擅長模仿人聲與環境音，羽毛為灰色，尾羽則是鮮紅色"},
        {L"帝雉", L"一千塊背面的鳥，是台灣體型最大的特有種鳥類"},
        {L"台灣冠八哥", L"本土原生物種，嘴部為牙白色，目前因外來種八哥競爭，野外族群數量大幅減少"},
        {L"家八哥", L"外來種，眼睛周圍有鮮黃色裸皮，常與白尾八哥成群出現，競爭力極強"},
        {L"白尾八哥", L"外來種，都市裡最常見的八哥，嘴部為橘黃色，飛行時翅膀與尾部可見明顯白斑"},
        {L"夜鷹", L"常在深夜發出極其響亮的「啾——、啾——」鳴叫聲，可謂《今夜無人入睡》（也許有人將這個作品翻譯成《今晚誰都別睡》）"},
        {L"ERROR", L"Oops□未找到相關資訊，請更換關鍵字再次搜尋。"}
    };

    wstring search_str;
```

```
    int size = 21;
    wcout << L"歡迎使用《我所知道的禽類》搜尋引擎□" << endl;
    wcout << L"請輸入您要查找的關鍵字：";
    getline(wcin, search_str);
    replying(database, size, search_str);
    return 0;
}
```

使用者將輸入一個欲搜尋的字串，程式將會在資料庫中比對該鳥類的名稱（第一欄）。若找到對應的名稱，則輸出該鳥類的名稱與介紹；若未能完全與使用者輸入的內容匹配，則輸出資料庫最後一筆的錯誤訊息□ERROR□□請注意本題所要求輸出的標點符號為中文全形。請完成名為 **bird.cpp** 與 **bird.h** 的 C++ 語言程式，其中分別包含 replying 函式的 **Implementation** 與其 **Prototype 宣告**□

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp bird.o
    g++ main.cpp bird.o
bird.o: bird.cpp bird.h
    g++ -c bird.cpp
clean:
    rm -f *.o *~ *.*~ a.out
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p7]▲./a.out↵
歡迎使用《我所知道的禽類》搜尋引擎！
請輸入您要查找的關鍵字：  亞洲輝椋鳥
亞洲輝椋鳥：城鎮和城市中大量分佈，牠們棲息於廢棄的建築物和樹木中，經常成群活動，被認為是最吵鬧的鳥類之一
[3:23▲user@ws▲p7]▲./a.out↵
歡迎使用《我所知道的禽類》搜尋引擎！
請輸入您要查找的關鍵字：  玄鳳鸚鵡
玄鳳鸚鵡：又稱雞尾鸚鵡，最明顯的特徵是頭頂的冠羽和臉頰兩側的橘色圓斑，看起來就像香蕉形狀的皮卡丘
[3:23▲user@ws▲p7]▲./a.out↵
歡迎使用《我所知道的禽類》搜尋引擎！
請輸入您要查找的關鍵字：  雪鴞
ERROR□Oops□未找到相關資訊，請更換關鍵字再次搜尋。
[3:23▲user@ws▲p7]▲./a.out↵
歡迎使用《我所知道的禽類》搜尋引擎！
請輸入您要查找的關鍵字：  斑鳩
珠頸斑鳩：頸部後方有一塊明顯的黑色白點斑塊，像是戴了珍珠項鍊，是都市裡最常見的「咕顧估」
灰斑鳩：全身羽毛呈灰淺褐色，頸後有一條細長的黑環，就像是脖子掛著頭戴式耳機一樣
[3:23▲user@ws▲p7]▲./a.out↵
歡迎使用《我所知道的禽類》搜尋引擎！
請輸入您要查找的關鍵字：  鳩
紅鳩：全身覆蓋著亮眼的酒紅色羽毛，頭部至頸部則呈鼠灰色，後頸有一道明顯的黑色頸環，就像是脖子掛著頭戴式耳機一樣
```

金背鳩：因爲翅膀覆羽具備顯眼的紅褐色或金棕色邊緣，在陽光下閃爍金色光芒而得名
珠頸斑鳩：頸部後方有一塊明顯的黑色白點斑塊，像是戴了珍珠項鍊，是都市裡最常見的「咕顧估」
灰斑鳩：全身羽毛呈灰淺褐色，頸後有一條細長的黑環，就像是脖子掛著頭戴式耳機一樣
鳩：英文俗名pigeons和「dove在中文可以譯成鴿或鳩，基本上，較小的叫dove較大的叫pigeon↵
[3:23▲user@ws▲p7]▲./a.out↵
歡迎使用《我所知道的禽類》搜尋引擎！
請輸入您要查找的關鍵字： 哥
台灣冠八哥：本土原生物種，嘴部為牙白色，目前因外來種八哥競爭，野外族群數量大幅減少
家八哥：外來種，眼睛周圍有鮮黃色裸皮，常與白尾八哥成群出現，競爭力極強
白尾八哥：外來種，都市裡最常見的八哥，嘴部為橘黃色，飛行時翅膀與尾部可見明顯白斑
[3:23▲user@ws▲p7]▲



在一個字串尋找子字串時，同學們可以使用定義在 string 或是 wstring 類別中的成員函式（member function）在物件導向的章節將會提及，find()找到子字串的首次出現在該字串中的位置，請參考下列程式碼：

```
string str = "Hello World";
cout << str.find("Wor") << endl; // 這裡將輸出「6 」
```

那如果沒有找到的話 str.find 會回傳什麼呢？在 <string> 類別中定義了一個名稱為（npos）特殊常數（constant），其意涵為（no position）未找到之意，請參考下列程式碼：

```
string str = "Hello World";
if(str.find("wii") == string::npos)
{
    cout << "Not Found!" << endl;
}
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 請注意本題所要求輸出的標點符號為中文全形。
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
  - bird.h
  - bird.cpp

*p9 凱撒密碼解碼器

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "caesar_cipher_decode.h"
#include <iostream>
using namespace std;

int main()
{
    string str;
    cout << "[Input cipher]: ";
    getline(cin, str);
    caesar_decode(str);
    cout << "[ Plaintext ]:  ";
    cout << str << endl;
}
```

凱撒密碼（Caesar Cipher）是一種簡單的加密系統，其運作原理是將兩個相同的字母表上下放置，並且錯動產生一個固定的偏移量。如下所示，當偏移量為 1 時，密文字母表上的每一個字元都是明文字母表中的上一個字元。如果遇到最後一個字母也就是 'Z'（'z'）時，其下一個字元重新回到 'A'（'a'）形成一個閉環。

```
明文字母表：ABCDEFGHIJKLMNOPQRSTUVWXYZ
密文字母表：ZABCDEFGHIJKLMNOPQRSTUVWXY -> 向後偏移一個字元
```

也就是說，明文的 'B' 對應到密文的 'A';明文的 'C' 對應到密文的 'B'...同理，小寫的英文字母字元也是以相同的方式運作，至於**數字與其他字元皆保持相同**。

請完成名為 **caesar_cipher_decode.cpp** 與 **caesar_cipher_decode.h** 的 C++ 語言程式，其中分別包含 caesar_decode 函式的 **Implementation** 與其 **Prototype 宣告**。

本題將提供一些密文讓同學們測試，在這些明文中將包含「stopdecoding」字串（明文僅包含一個「stopdecoding」字串），如果密文不斷解碼直到「stopdecoding」出現，則停止解碼。

```
密文1:Qgm escw ew ydgo. Tml A ugnwj mh, ogf'l dwl al kzgo. Kg A'e hmllafy eq vwxwfkwk mh. 'Usmkw A vgf'l osffs xsdd af dgnw. Ax A wnwj vav lzsl, A lzafc A'v zsnw s zwsjl sllsuc klghvwugvafy.

密文2:S rokb iyeb rokbdlokd dy dro lokd yp dro nbewc. Yr, grkd k crkwo drkd iye mkwo robo gsdr cywoyxo cdyznomynsxq. Cy grsvo iye'bo robo sx wi kbwc. Vod'c wkuo dro wycd yp dro xsqrd vsuo go'bo qyxxk nso iyexq.

密文3:Fm fm fm fm fmfmpefy. Alex wxsthigshmrk alex alex alex alex xli hsk hsmrk.
```

```
密文4:Kpk fvb ylzjbl h mve vu h zuvdf tvbuahpu? Hyl fvb aol dopal mve? Uv!
P't aol Zwpjf Zhsalk Kbjr zavwkljvkpun!

密文5:bcxymnlxmrwp qccyb://fff.hxdcdkn.lxv/fjclq?e=_LnSIN7wE_t

密文6:uggcf://jjj.ovyvovyv.pbz/ivqrb/OI1AZ4z1E7U8/?c=2&funer_fbhepr=pbcl_jro
fgbcqrpbqvat
```

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp caesar_cipher_decode.o
    g++ main.cpp caesar_cipher_decode.o
caesar_cipher_decode.o: caesar_cipher_decode.cpp caesar_cipher_decode.h
    g++ -c caesar_cipher_decode.cpp
clean:
    rm -f *.o *~ *.*~ a.out
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p9]▲./a.out↵
[Input▲cipher]:▲Qgm▲escw▲ew▲ydgo.▲Tml▲A▲ugnwj▲mh,▲ogf'l▲dwl▲al▲kzgo.▲Kg▲A'e▲hm
llafy▲eq▲vwxwfkwk▲mh.▲'Usmkw▲A▲vgf'l▲osffs▲xsdd▲af▲dgnw.▲Ax▲A▲wnwj▲vav▲lz
sl,▲A▲lzafc▲A'v▲zsnw▲s▲zwsjl▲sllsuc▲klghvwugvafy.↵
[▲Plaintext▲]:▲▲You▲make▲me▲glow.▲But▲l▲cover▲up,▲won't▲let▲it▲show.▲So▲I'm▲putting▲my▲defen
ses▲up.▲'Cause▲I▲don't▲wanna▲fall▲in▲love.▲If▲I▲ever▲did▲that,▲I▲think▲I'd▲have▲a▲heart▲attack▲st
opdecoding.↵
[3:23▲user@ws▲p9]▲./a.out↵
[Input▲cipher]:▲S▲rokb▲iyeb▲rokbdlokd▲dy▲dro▲lokd▲yp▲dro▲nbewc.▲Yr,▲grkd▲k▲crk
wo▲drkd▲iye▲mkwo▲robo▲gsdr▲cywoyxo▲cdyznomynsxq.▲Cy▲grsvo▲iye'bo▲robo▲sx▲
wi▲kbwc.▲Vod'c▲wkuo▲dro▲wycd▲yp▲dro▲xsqrd▲vsuo▲go'bo▲qyxxk▲nso▲iyexq.↵
[▲Plaintext▲]:▲▲I▲hear▲your▲heartbeat▲to▲the▲beat▲of▲the▲drums.▲Oh,▲what▲a▲shame▲that▲you▲c
ame▲here▲with▲someone▲stopdecoding.▲So▲while▲you're▲here▲in▲my▲arms.▲Let's▲make▲the▲mos
t▲of▲the▲night▲like▲we're▲gonna▲die▲young.↵
[3:23▲user@ws▲p9]▲
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
  - caesar_cipher_decode.h
  - caesar_cipher_decode.cpp

*p10 高級水果攤

請參考下列的 main.cpp 程式（請依據檔名旁的路徑至 ws 取得程式碼）：

```
#include "fruit_products.h"
#include <iostream>
#define SIZE 21
using namespace std;

int main()
{
    string user_input = "";
    ProductInfo found_product;
    ProductInfo recommend_product;
    ProductInfo fruit_product[] = {
        {"Studio Display XDR", 94500, MONITOR},
        {"MacBook Air 13\"", 35900, LAPTOP},
        {"MacBook Air 15\"", 42900, LAPTOP},
        {"MacBook Pro 14\"", 54900, LAPTOP},
        {"MacBook Pro 16\"", 54900, LAPTOP},
        {"Studio Display", 52900, MONITOR},
        {"iPhone 17 Pro", 39900, PHONE},
        {"iPad Pro 11\"", 29400, TABLET},
        {"iPad Pro 13\"", 40400, TABLET},
        {"iPad Air 11\"", 18200, TABLET},
        {"iPad Air 13\"", 25200, TABLET},
        {"MacBook Neo", 19900, LAPTOP},
        {"iPhone Air", 36900, PHONE},
        {"Mac Studio", 67900, DESKTOP},
        {"iPhone 17e", 21900, PHONE},
        {"iPhone 16e", 20900, PHONE},
        {"iPhone 17", 29900, PHONE},
        {"iPhone 16", 25900, PHONE},
        {"iPhone 15", 20900, PHONE},
        {"Mac mini", 19900, DESKTOP},
        {"iMac", 42900, DESKTOP}
    };

    cout << "Welcome to Premium Fruit Stand." << endl;
    cout << "What can I get for you? ";
    getline(cin, user_input);
    if (!is_in_stock(fruit_product, SIZE, user_input, &found_product,
&recommend_product))
    {
        cout << "Sorry! The product you entered was not found or currently
not available." << endl;
```

```
        return -1;
    }

    cout << "Product Found: " << found_product.name << " Price: " <<
found_product.price << endl;
    cout << "You can also consider: " << recommend_product.name << " Price:
" << recommend_product.price << endl;
    return 0;
}
```

使用者將輸入**包含一個產品名稱的字串**，請檢查該產品是否存在於 main.cpp 中的 fruit_product 陣列，並向使用者推薦同類產品中價格最接近使用者要尋找的產品（若存在多個「價格最接近使用者要尋找的產品」則向使用者推薦 fruit_product 陣列中索引值最小（首個匹配）的產品。請完成名為 **fruit_products.cpp** 與 **fruit_products.h** 的 C++ 語言程式，其中分別包含 is_in_stock 函式的 **Implementation** 與其 **Prototype 宣告**。

相關的使用者自訂資料型態請參考如下 required.h 檔案，請於「fruit_products.h」檔案中載入 required.h 標頭檔。

```
#include <string>
using namespace std;

enum ProductCategory
{
    MONITOR,
    LAPTOP,
    DESKTOP,
    TABLET,
    PHONE
};

struct ProductInfo
{
    string name;
    int price;
    ProductCategory category;
};
```

本題的相關程式將使用以下的 Makefile 進行編譯：

```
all: main.cpp fruit_products.o
    g++ main.cpp fruit_products.o
fruit_products.o: fruit_products.cpp fruit_products.h
    g++ -c fruit_products.cpp
clean:
    rm -f *.o *~ *.*~ a.out
```

此題的執行結果可參考如下：

```
[3:23▲user@ws▲p10]▲./a.out↵
Welcome▲to▲Premium▲Fruit▲Stand.↵
What▲can▲I▲get▲for▲you?▲iphone▲17↵
Sorry!▲The▲product▲you▲entered▲was▲not▲found▲or▲currently▲not▲available.↵
[3:23▲user@ws▲p10]▲./a.out↵
Welcome▲to▲Premium▲Fruit▲Stand.↵
What▲can▲I▲get▲for▲you?▲iPhone▲17↵
Product▲Found:▲iPhone▲17▲Price:▲29900↵
You▲can▲also▲consider:▲iPhone▲16▲Price:▲25900↵
[3:23▲user@ws▲p10]▲./a.out↵
Welcome▲to▲Premium▲Fruit▲Stand.↵
What▲can▲I▲get▲for▲you?▲I▲want▲Mac▲Studio!!!↵
Product▲Found:▲Mac▲Studio▲Price:▲67900↵
You▲can▲also▲consider:▲iMac▲Price:▲42900↵
[3:23▲user@ws▲p10]▲./a.out↵
Welcome▲to▲Premium▲Fruit▲Stand.↵
What▲can▲I▲get▲for▲you?▲the▲cheapest▲phone,▲iPhone▲15↵
Product▲Found:▲iPhone▲15▲Price:▲20900↵
You▲can▲also▲consider:▲iPhone▲16e▲Price:▲20900↵
[3:23▲user@ws▲p10]▲./a.out↵
Welcome▲to▲Premium▲Fruit▲Stand.↵
What▲can▲I▲get▲for▲you?▲silver▲MacBook▲Neo↵
Product▲Found:▲MacBook▲Neo▲Price:▲19900↵
You▲can▲also▲consider:▲MacBook▲Air▲13"▲Price:▲35900↵
[3:23▲user@ws▲p10]▲
```



在一個字串尋找子字串時，同學們可以使用定義在 string 或是 wstring 類別中的成員函式（member function）在物件導向的章節將會提及，find()找到子字串的首次出現在該字串中的位置，請參考下列程式碼：

```
string str = "Hello World";
cout << str.find("Wor") << endl; // 這裡將輸出「6 」
```

那如果沒有找到的話 str.find 會回傳什麼呢？在 <string> 類別中定義了一個名稱為 npos 特殊常數（constant），其意涵為「no position」未找到之意，請參考下列程式碼：

```
string str = "Hello World";
if(str.find("wii") == string::npos)
```

```
{
    cout << "Not Found!" << endl;
}
```

- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 請於fruit_products.h檔案中載入 required.h 標頭檔
- 本題若有使用浮點數的需求，請使用 double 型態。
- 本題應繳交檔案如下（至於 main.c 與 Makefile 則不需繳交）：
  - fruit_products.h
  - fruit_products.cpp

From:
https://junwu.nptu.edu.tw/dokuwiki/ - **Jun Wu的教學網頁**
國立屏東大學資訊工程學系
CSIE, NPTU
Total: 343751

Permanent link:
https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=cpp:2026spring:hw5

Last update: **2026/04/22 09:03**