

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

作業一、

Home Works			StringCls
<ul style="list-style-type: none"><li>• Make double pointer matrix</li><li>• Users can define size <math>m, n</math></li><li>• Assign values by random</li><li>• Print matrix</li><li>• Finish your String Class</li><li>• Complete FindString() 搜尋字串 (需完成，任何方法皆可)</li><li>• Ref: P114~P119</li><li>• Watch git lecture video</li><li>• branch and merge</li></ul>	建構子	+<<constructor>>	StringCls()
	解構子	+<<destructor>>	~StringCls()
	複製字串	+strCPY(src:char*):void	
	串接字串	+strCAT(src:char*):void	
	回傳物件中的字串長度	+strLen():unsigned int	
	取得物件中的字串	+GetString():char*	
	顯示字串	+PrintString():int	
	多載運算子=	+FindString(pat:StringCls):int	
	多載運算子+	+operator=(src:StringCls):StringCls	
		+operator+(src:StringCls):StringCls	
	字串內容	-data:char*	
	字串長度	-len:unsigned int	

執行流程：

```
矩陣類型[m]x[n]
請輸入陣列[m]值:3
請輸入陣列[n]值:4
[[3 7 7 3]
 [1 6 5 4]
 [3 2 8 7]]
str 新插入的字串:apple
回傳字串str的長度:5
=====串聯b後=====
apple pen
設定新class str2:
PPAP
將str+str2:apple pen PPAP
在添加後的字串搜尋PPAP
有包含字串
將str等化為str2後,str:PPAP
PS D:\vscode>
```

班級：醫工二甲  
學號：11125107  
姓名：李俞憲  
說明：

```
int main(){
    int **num;//double ptr
    int m=0;//列
    int n=0;//高
    cout<<"矩陣類型[m]x[n]"<<endl;
    cout<<"請輸入陣列[m]值:";
    cin>>m;
    cout<<"請輸入陣列[n]值:";
    cin>>n;
```

首先，利用動態記憶體配置，先讓使用者輸入相應的二維矩陣大小

```
num = new int*[m]; //生成m個arr

// 創建二維列
int** two_dimensional_array = new int*[m]; //利用雙指標先創建列
for (int i = 0; i < m; i++)
{
    two_dimensional_array[i] = new int[n]; //使用for迴圈，將行補齊
}

//插入亂數
srand( time(NULL) );
for (int i = 0; i < m; i++)
{
    for (int j = 0; j < n; j++)
    {
        two_dimensional_array[i][j] = rand() % 10;
    }
}
```

利用雙重指標的特性，先創建列[m]，在使用 for 迴圈將行[n]補齊，並插入亂數。

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
//顯示亂數
cout<<"[";
for (int i=0;i<m;i++)//m
{
    cout<<"[";
    for (int j=0;j<n;j++)//n
    {
        if (j == n-1)
        {
            cout << two_dimensional_array[i][j];
        }
        else
        {
            cout << two_dimensional_array[i][j] << " ";
        }
    }
    if (i == m-1)
    {
        cout<<"]]"<<endl;
    }
    else
    {
        cout<<"]\n";
    }
}
```

最後將陣列內容逐一輸出，為了美觀，我在內部使用 if 判斷式，將每一列的頭尾加上中括號，並分段。

```
for (int i = 0; i < m; i++)
{
    delete[] two_dimensional_array[i];
}
delete[] two_dimensional_array;
```

執行完畢後仍需使用 for 迴圈，將每一行[m]逐一刪除，可將其視為多個指標。

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
//class
StringClassd str;
StringClassd str2;
char a[] = "apple";
char b[] = "pen";
char c[] = "PPAP";
str.strCPY(a);
cout<<"str 新插入的字串:";
str.PrintString();
cout<<"回傳字串str的長度:"<<str.strLEN()<<endl;
cout<<"====串聯b後===="<<endl;
str.strCAT(b);
str.PrintString();
cout<<"設定新class str2:"<<endl;
str2.strCPY(c);
str2.PrintString();
cout<<"將str+str2:";
```

接下來是 class 的作業(前半段和之前作業一樣)，我先設定 3 個字串，並分別利用 class 內部的 function，但這次的 operator = 應該為 void 型態(和上次作業一樣)，不應該是回傳一組 class，這樣可能無從下手...

```
cout<<"在添加後的字串搜尋PPAP"<<endl;
char find[] = "PPAP";
if (str_add.FindString(find))
{
    cout<<"有包含字串\n";
}
else
{
    cout<<"查無字串"<<endl;
}
//cout<<"回傳值為:"<<str2.FindString(find)<<endl;
cout<<"將str等化為str2後,str:";
str = str2;
str.PrintString();
return 0;
```

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

這次有要求學生需要打出搜尋字串的 function，因為回傳值為 int 型態，我預設如果有查詢到字串就回傳 1(True)，若無，則回傳 0(False)，接下來會針對該 function 進行詳述的介紹。

```
int StringClassd::FindString(char* src)//搜尋字串
{
    int counter =0;
    int goal_point = strlen(src) ;
    while (true)
    {
        if ((strlen(src)+counter) > len)//比對次數用盡時
        {
            return 0;//回傳False
        }
    }
}
```

我的想法是使用線性的比對方式，來找出目標字串有無存在，counter 是紀錄目標字串在進行位移時的差值，用於修正 index 以及判斷可執行比對的最大次數，goal\_point 是指當目標字串完全符合時的條件。

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
int point = 0;
for (int i=0;i<len;i++)
{
    //cout<<"第"<<i+1<<"次比對"<<endl;
    if (data[i] == src[i-counter])
    {
        point++;
    }
    else
    {
        counter++;
        //cout<<"無效，跳脫迴圈\n";
        continue;
    }

    if (point == goal_point)//較驗是否完成比對
    {
        return 1;//回傳True;
    }
}

//cout<<"錯誤\n";
}
```

point 是指當比對的每一組內容，如果相同時則加一分，直到 point==goal\_point 時，或是比對次數用盡時，如果出現不吻合的結果時，程式挑出迴圈，將 point 重新歸零並將 counter 差值加一，重新比對內容，最 return 的值採用布林的類型以 0、1 表示有無搜尋到字串，其實也可以改為當出現吻合的字串內容時，回傳第一個的 index 值，若無，則回傳 0 or -1 等方式。

程式碼：

//main

```
//#include <iostream>
#include "work1.h"
#include "work1_function.cpp"
```

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
using namespace std;

int main(){
    int **num;//double ptr
    int m=0;//列
    int n=0;//高
    cout<<"矩陣類型[m]x[n]"<<endl;
    cout<<"請輸入陣列[m]值:";
    cin>>m;
    cout<<"請輸入陣列[n]值:";
    cin>>n;

    num = new int*[m]; //生成 m 個 arr

    // 創建二維列
    int** two_dimensional_array = new int*[m]; //利用雙指標先創建列
    for (int i = 0; i < m; i++)
    {
        two_dimensional_array[i] = new int[n]; //使用 for 迴圈，將行補齊
    }

    //插入亂數
    srand( time(NULL) );
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            two_dimensional_array[i][j] = rand() % 10;
        }
    }

    //顯示亂數
    cout<<"[";
    for (int i=0;i<m;i++)//m
    {
        cout<<"[";
        for (int j=0;j<n;j++)//n
        {
```

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
        if (j == n-1)
        {
            cout << two_dimensional_array[i][j];
        }
        else
        {
            cout << two_dimensional_array[i][j] << " ";
        }
    }
    if (i == m-1)
    {
        cout<<"]]"<<endl;
    }
    else
    {
        cout<<"]]"<<endl;
    }
}

for (int i = 0; i < m; i++)
{
    delete[] two_dimensional_array[i];
}
delete[] two_dimensional_array;

//class
StringClassd str;
StringClassd str2;
char a[] ="apple";
char b[] ="pen";
char c[] ="PPAP";
str.strCPY(a);
cout<<"str 新插入的字串:";
str.PrintString();
cout<<"回傳字串 str 的長度:"<<str.strLEN()<<endl;
```



班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
    cout<<"=====串聯 b 後===== "<<endl;
    str.strCAT(b);
    str.PrintString();
    cout<<"設定新 class str2:"<<endl;
    str2.strCPY(c);
    str2.PrintString();
    cout<<"將 str+str2:";

    StringClassd str_add;
    str_add = (str + str2);
    str_add.PrintString();

    cout<<"在添加後的字串搜尋 PPAP"<<endl;
    char find[] = "PPAP";
    if (str_add.FindString(find))
    {
        cout<<"有包含字串\n";
    }
    else
    {
        cout<<"查無字串"<<endl;
    }
    //cout<<"回傳值為:"<<str2.FindString(find)<<endl;
    cout<<"將 str 等化為 str2 後,str:";
    str = str2;
    str.PrintString();
    return 0;
}
```

//.h

```
#include <iostream>
#include<cstdlib>
#include<cstring>
#include <typeinfo>
#include <stdlib.h> /* 亂數相關函數 */
#include <time.h> /* 時間相關函數 */
using namespace std;
#pragma once
```

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
class StringClassd{
private://私有
    char* data;//字串內容
    unsigned int len;//字串長度
public://公有
    StringClassd();//建構子
    ~StringClassd();//解構子
    void strCPY(char*);//複製 s
    void strCAT(char*);//串接
    unsigned int strLEN();//回傳字串長度
    char *GetString();//取得字串
    int FindString(char*);//搜尋字串
    int PrintString();//輸出
    void operator=(StringClassd src);//等號(class=class)
    StringClassd operator+(StringClassd src);
};
```

//.cpp

```
#include "work1.h"

StringClassd::StringClassd()
{
    data = NULL;
    len=0;
}

StringClassd::~~StringClassd()
{
    //delete data; //原來是 delete 在搞鬼!
    data = nullptr;
}
```

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
void StringClassd::strCPY(char* src) //複製
{
    if (data != NULL) {
        delete data;
        data =NULL;
    }
    data = new char[strlen(src) + 1]; //給予動態記憶體配置
    strcpy(data, src);
    len = strlen(data); //不含\0
}

void StringClassd::strCAT(char* str) //串接
{
    int input_len;
    bool flag;
    int temp=0;
    input_len = strlen(str);
    //cout<<"input_len:"<<input_len<<endl;
    char* char_plus = new char[(len+input_len+2)]; //有兩組'\0'
    flag = true; //判斷是否為最後一個結尾"\0" T:第一段 F:第二段
    for (int i=0; i<(input_len+len+2); i++)
    {
        if (flag && data[i] != '\0')
        {
            char_plus[i]=data[i];
        }
        else if (flag && data[i] == '\0')
        {
            char_plus[i]=' '; //在此把第一段的'\0'更改為空格
            flag = false; //表示進入第二段
            //cout<<"Part2"<<endl;
            temp = i+1;
        }
        else
        {
            char_plus[i]=str[i-temp]; //第二段內容一樣直接 copy 上去(含\0)
        }
    }
}
```

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
    }

    //delete[] data;//完成後清除舊資料
    //data = NULL;
    len = strlen(char_plus);
    data = char_plus; //將指標轉向新陣列
    return;
}

unsigned StringClassd::strLEN()//回傳字串長度
{
    int len_cpy=0;
    len_cpy = len;
    return len_cpy;
}

char* StringClassd::GetString()//取得字串
{
    char* data2pub = new char[len+1];
    strcpy(data2pub,data);
    return data2pub;
}

int StringClassd::FindString(char* src)//搜尋字串
{
    int counter =0;
    int goal_point = strlen(src) ;
    while (true)
    {
        if ((strlen(src)+counter) > len)//比對次數用盡時
        {
            return 0;//回傳 False
        }

        int point = 0;
        for (int i=0;i<len;i++)
        {
            //cout<<"第"<<i+1<<"次比對"<<endl;
```

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
        if (data[i] == src[i-counter])
        {
            point++;
        }
        else
        {
            counter++;
            //cout<<"無效，跳脫迴圈\n";
            continue;
        }

        if (point == goal_point)//較驗是否完成比對
        {
            return 1;//回傳 True;
        }
    }
    //cout<<"錯誤\n";
}

int StringClassd::PrintString()
{
    for(int i=0;i<len;i++)
    {
        char* ptr;
        ptr = GetString();
        cout<<ptr[i];
    }
    cout<<endl;
    //cout<<"len:"<<len<<endl;//驗證
    return 0;
}

void StringClassd::operator=(StringClassd src)//A=B
{
    if (data != NULL)
```

班級：醫工二甲  
學號：11125107  
姓名：李俞憲

```
{
    delete data; //刪除舊資料
    data = NULL;
}

len = src.len;
data = new char[(len+1)];
strcpy(data,src.data);
return;
}

StringClassd StringClassd::operator+(StringClassd src)//加號"本身"與"src"不會改變
{
    StringClassd temp;//創建一組 class
    temp.len = len;//將 len 值相加入 temp
    temp.data = new char[(temp.len)];

    strcpy(temp.data, data); // 複製 this 的內容
    temp.strCAT(src.data); //串接上 src 的內容

    return temp;
}
```

---

### 補充說明（遇到的困難或心得，選填）：

雙指標提供了相當便利的功能，不過在使用時仍需要相當謹慎，我會使用紙筆紀錄指標的指向，以免錯誤的使用，這次作業有搜尋字串的功能，雖然課堂有講解 KMP 算法，但是我還需要一點時間再想看看，所以採用了線性的暴力搜尋法，最後上次的作業再解構子中 delete data; 時會出現例外狀況，這次我將它註解掉了，程式可以正常運行，其他使用 vscode 的同學也有出現一樣的問題，對此感到不解。