解題說明

在這題需要分成兩段來理解

要知道 ADT 跟 ”私有變數” 這兩個事情

ADT(abstract data type)

抽象資料類型，處於一種更底層的資料結構，可以想像成我做的 class 裡面的設定之變數，要修改/新增該數值必須宣告一個該 class 的變數才行。如圖(1)，這裡說明了該 Term 類別擁有 coef,exp 這兩個變數，圖(2)則為陣列內輸入各項值。

```
class Term { // 在這裡表示多項式一個一個的數值
public:
    float coef;  // 前方係數
    int exp;     // 次方數

    Term(float c = 0.0f, int e = 0) : coef(c), exp(e) {}

    void Print(){ // 列出單個的數值
        if (exp == 0) {
            cout << coef;
        } else {
            cout << coef << "x^" << exp;
        }
    }
};
```

圖(1)

```
vector<Term> terms1 = {Term(3, 0), Term(2, 1), Term(1, 2)};
vector<Term> terms2 = {Term(4, 0), Term(2, 1)};
```

圖(2)

公有私有成員

適用於類別類型的成員。類別類型的每個成員都有稱為存取等級的屬性，該屬性決定誰可以存取該成員。

C++有三種不同的存取等級：公用（public）、私有（private）和受保護（protected）。每當存取成員時，編譯器都會檢查該成員的存取級別，是否允許存取該成員。如果不允許訪問，編譯器將產生編譯錯誤。圖(3)是對照表。

圖(4)(5)為範例。

訪問級別	訪問說明符	會員可訪問	子類別可訪問	public可訪問
公有	public:	是	是	是
受保護	protected:	是	是	否
私人	private:	是	否	否

圖(3)

```
class Date
{
// 这里的所有成員都是 private

public: // 這裡是 public:

    void print() const // public
    {
        // public可以訪問 private 成員
        std::cout << m_year << '/' << m_month << '/' << m_day;
    }

private: // 這裡是 private:

    int m_year { 2020 };  // private
    int m_month { 14 }; // private
    int m_day { 10 };   // private
};
```

圖(4)

```
int main()
{
    Date d{};
    d.print();  // 非Datet成員 可以訪問 Date的public 成員

    return 0;
}
```

圖(5)

Algorithm Design & Programming

參閱程式檔案 ”APT.cpp”

效能分析

時間複雜度

加法

O(n*m) n = array1.length m = array2.length

乘法

O(n*m*r) n = array1.length m = array2.length r = result.length

空間複雜度

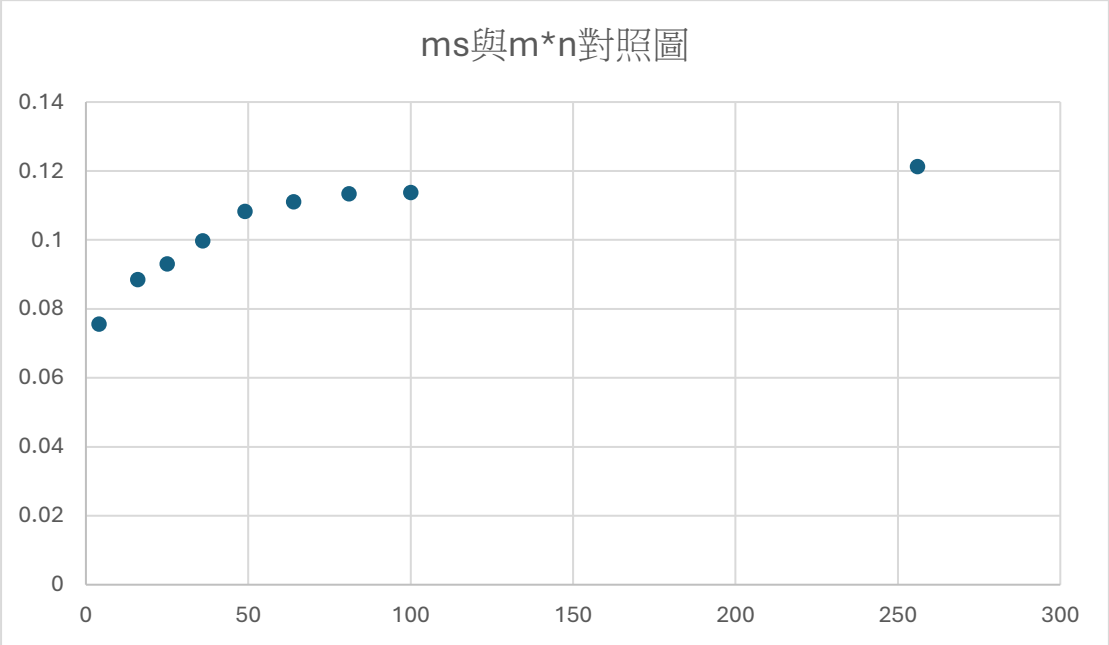
O(2n) n = term 總數

測試與驗證

```
vector<Term> terms1 = {Term(3, 0), Term(2, 1), Term(1, 2)};
vector<Term> terms2 = {Term(4, 0), Term(2, 1)};

Polynomial p1(terms1); //3 + 2x + 1x^2
Polynomial p2(terms2); //4 + 2x

Polynomial p3 = p1.Add(p2);
Polynomial p4 = p1.Mult(p2);

p1.Print(); // 輸出： 3 2x^1 +1x^2
p2.Print(); // 輸出： 4 2x^1
p3.Print(); // 輸出： 7 4x^1 +1x^2
p4.Print(); // 輸出： 12 14x^1 +8x^2 +2x^3

cout << "p1 在 x = 2 的值：" << p1.Eval(2) << endl;

return 0;
```

```
3 +2x^1 +1x^2
4 +2x^1
7 +4x^1 +1x^2
12 +14x^1 +8x^2 +2x^3
p1 在 x = 2 的值：11
```

效能量測



心得

主要是抽象資料的問題，剛開始寫時，沒啥太大概念，整個不知道如何下手，將自己想法寫出。好險有課本才不至於花太長的時間，關於 public , private 的事，理解了將副程式包裝的用處。主要就這幾點，學習到了，以上。