資料結構—HW2

作者:

黃昱翔

學號

40932626

December 1, 2024

Contents

Chapter 1

解題說明

透過重載運算子“>>”和“<<”，使多項式的輸入與輸出符合人類直觀格式。輸入時，逐項填入係數與指數；輸出時，格式化為數學表示法。

Chapter 2

程式實作

```
#ifndef POLYNOMIAL_H
#define POLYNOMIAL_H

#include <iostream>
using namespace std;

class Polynomial; // Forward declaration

class Term {
    friend class Polynomial;
    friend ostream& operator<<(ostream& os, const Polynomial& poly); // Output operator
    friend istream& operator>>(istream& is, Polynomial& poly);       // Input operator

private:
    float coef; // Coefficient
    int exp;    // Exponent
};
```

```
class Polynomial {
public:
    Polynomial(int capacity = 10); // Constructor with default capacity
    ~Polynomial();                 // Destructor

    friend ostream& operator<<(ostream& os, const Polynomial& poly); // Output operator
    friend istream& operator>>(istream& is, Polynomial& poly);       // Input operator

private:
    Term* termArray; // Array of non-zero terms
    int capacity;    // Size of the term array
    int terms;       // Number of non-zero terms

    void Expand();   // Expand the capacity of the term array
};
```

Chapter 3

效能分析

時間複雜度

輸入:**O**(n)，其中 n 為多項式的非零項數

輸出:**O**(n)

空間複雜度

僅需常數額外空間 **O**(1)

Chapter 4

測試與驗證

測試輸入: $P(x) = 3X^2+2X+1$

驗證輸出: 格式化為 $3X^2+2X+1$

效能量測

測試輸入與輸出的性能，當多項式的項數越來越大時，是否能夠在合理時間內完成。

Chapter 5

心得討論

學會了透過重載運算子實現自定義輸入與輸出，在改進方面上可以將運算子設計成更廣泛的格式，以支援更多的輸出樣式