자료구조 오픈랩2

컴퓨터공학과 20171646 벅태윤

|  |  |
| --- | --- |
| 스크린샷, 노트북, 컴퓨터, 모니터이(가) 표시된 사진  자동 생성된 설명 | 스크린샷, 노트북, 컴퓨터, 모니터이(가) 표시된 사진  자동 생성된 설명 |
|  |  |

스크린샷, 그리기이(가) 표시된 사진

자동 생성된 설명

그림과 같이 다항식의 계수와 지수를 나타내는 A.txt와 B.txt라는 텍스트파일을 만들어 코드를 실행을 시켰고 다음과 같은 결과를 얻었다.

(코드 / 알고리즘 설명)

다항식의 계수와 지수를 나타내기 위해 다음과 같은 구조체를 선언하였고 polynomial type 배열인 terms와 int형 변수 avail을 전역변수로 선언하였다.

스크린샷, 노트북, 컴퓨터, 모니터이(가) 표시된 사진

자동 생성된 설명

main에서는 fpa와 fpb라는 FILE변수를 선언하여 각각 “A.txt”, “B.txt”를 읽어 들였다. “A.txt”의 값부터 넣어주는데, i라는 변수를 0에서부터 하나씩 증가시키면서 terms[i].coef에 계수를, terms[i].expon에 지수를 넣어주었다. 이후 A식의 끝을 표현하는 finisha에 i-1이라는 값을 넣어주었다. (starta는 선언할 때 0값을 넣어주었다.) 다음에 B식의 계수와 지수 값을 terms배열에 넣어준 이후 B식의 시작과 끝을 표현하는 startb와 finishb에 값을 넣어주었다. 이후 이후 avail은 finishb다음으로 지정하였다.

스크린샷, 노트북, 컴퓨터, 실내이(가) 표시된 사진

자동 생성된 설명

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| starta | ~~~~ | finisha | startb | ~~~~~ | finishb | avail |
| A식 계수 | | | B식 계수 | | |  |
| A식 지수 | | | B식 지수 | | |

표와 같이 terms배열에 저장된 위치를 표현하는 starta, startb, finisha, finishb와 (A+B)를 계산한 결과를 저장하기 시작할 위치인 avail을 지정한 이후 padd함수를 호출한다. padd함수가 받는 변수 중 startd와 finishd는 각각 계산한 식이 terms배열에 저장될 위치의 처음과 끝을 나타낸다.

padd함수는 (A+B)의 결과를 terms배열에 startd부터 finishd까지 저장하는 역할을 한다. avail부터 계산된 식을 넣어주어야 하기 때문에 startd는 avail로 초기화 시킨다. padd함수 내에서 쓰이는 함수는 compare함수, attach함수가 있는데, compare함수는 integer형 변수 2개를 받아 값을 비교하여 각 결과마다 1 or 0 or -1을 return해주는 역할을 한다. attach함수는 계산된 결과를 terms배열에 넣어주면서 한 번 넣어줄 때마다 avail을 다음 위치로 옮겨주는 역할을 한다.

|  |  |
| --- | --- |
| 스크린샷, 노트북, 컴퓨터, 실내이(가) 표시된 사진  자동 생성된 설명 | 스크린샷, 노트북, 컴퓨터, 실내이(가) 표시된 사진  자동 생성된 설명 |

padd는 A식과 B식의 처음과 끝을 알려주는 변수를 받았기 때문에 이 함수 내에서 A식과 B식을 동시에 읽어들인다. 각각 starta와 startb부터 시작해서 compare함수를 통해 지수를 나타내는 terms[starta].expon, terms[startb].expon을 비교하는데, A의 지수가 크면 현재 starta가 가리키고 있는 지수와 계수 값을 avail이 가리키고 있는 위치에 넣어주고 avail과 starta를 1씩 증가시킨다. B의 지수가 크면 마찬가지로 startb가 가리키고 있는 값을 넣어주고 똑같이 avail과 startb를 1씩 증가시킨다. 비교를 하다보면 terms[starta].expon와 terms[startb].expon가 같은 경우가 생기는데 이때는 terms[starta].coef와 terms[startb].coef를 더한 값을 integer형 변수 coefficient에 넣어준 후 coefficient와 그에 해당하는 지수를 attach함수에 넘겨준다. 이후 starta와 startb모두 1씩 증가시킨다. 이 과정을 (starta <= finisha && startb <= finishb)라면 계속 반복을 하는데, 이는 즉 A 혹은 B식에서 더 계산을 할 수 있는 항이 존재하지 않을 때까지 위의 과정을 반복하라는 의미이다. 이후 while문을 탈출하면 A또는 B식에서 계산과정을 거치치 않은 항이 존재할 수 있다. 예를 들어

A = 2x^2 + x + 1

B = 4x^2 + 3x + 2

와 같은 식이라면 위의 과정을 통해 모든 항의 계산이 완료되지만

A = x^100 + x^99 + x^98

B = x^2 + x + 1

와 같은 식이라면 A식의 항만 넣다가 반복문이 끝나버리고 B식의 항들이 남아버리게 된다. 때문에 while문이 끝난 이후 남아있는 항들을 처리를 해주어야 하는데 이 때는 A식의 항만 남아있거나 B식의 항만 남아있고 모든 항들이 while문을 거쳐 저장이 된 항들보다 지수가 낮을 것이기 때문에 이 때는 별도의 과정 없이 attach함수를 통해 저장만 해주면 된다. 마지막으로 계산된 식의 끝을 나타내는 finishd에 avail-1값을 넣으면 다음의 표와 같아진다.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| starta | ~~~~ | finisha | startb | ~~~~~ | finishb | startd | ~~~~ | finishd | avail |
| A식 계수 | | | B식 계수 | | | (A+B)식 계수 | | |  |
| A식 지수 | | | B식 지수 | | | (A+B)식 지수 | | |

스크린샷, 노트북, 컴퓨터, 실내이(가) 표시된 사진

자동 생성된 설명

이후 다음의 코드와 같이 startd부터 finishd까지 계수와 지수를 출력시켜주고 fga와 fgb를 닫아주면 된다.