

프로그래밍 과제 II

과제 I에서 작성한 C 프로그램을 아래와 같이 수정하시오.

수정: $A(x) * B(x) = D(x)$ 일 때, $D(x)$ 를 과제 I의 항 배열 대신에 **linked list**에 저장.

과제 I의 C 코드 재사용 및 수정:

$A(x)$ 의 1개 항 ax^i 과 $B(x)$ 의 1개 항 bx^j 을 곱해서 항 abx^{i+j} 를 산출했을 때, abx^{i+j} 를 linked list에 반영하는 과정 및 최종 $D(x)$ 의 출력은 기존 코드를 수정 (즉, linked list 버전으로 대체) 나머지 코드 (ax^i 과 bx^j 을 곱해서 abx^{i+j} 를 산출하는 과정 포함) 는 과제 I의 코드 재사용

과제 II의 채점 범위: $D(x)$ 에 대한 코딩 부분 및 레포트

구현 요건:

- linked list의 자료형 선언: 반드시 수업시간에 설명한 교재의 C 코딩 방식을 사용한다. (다른 방식의 코딩도 가능하지만 사용하는 컴파일러에 따라 오류 발생할 수 있음)
- $D(x)$ 의 각 노드는 계수가 0이 아닌 각 항을 표현한다.
- $D(x)$ 를 저장하는 linked list는 헤더가 없는 singly linked list (교재의 또다른 용어로 chain이라고 부르는 형태)로 한다.
- $D(x)$ 를 저장하는 linked list에서 노드의 순서는 지수 내림순으로 한다. ($A(x)$ 의 1개 항 ax^i 과 $B(x)$ 의 1개 항 bx^j 을 곱해서 항 abx^{i+j} 를 산출했을 때, abx^{i+j} 를 linked list에 바로 반영한다. 1개의 항 abx^{i+j} 를 linked list에 반영할 때마다 그 결과 linked list의 노드 순서는 지수 내림순 유지)
- $A(x)$ 와 $B(x)$ 가 과제 I의 계수 배열이 아니라 linked list로 주어지는 경우에는 **0점 처리** (과제 I의 코드에서 $A(x)$ 와 $B(x)$ 의 입력, 저장, 항 조회, 출력 등에 관한 부분의 일부 수정은 필요시 가능. 그 경우 레포트에 수정 내용 및 사유 설명. 단, $A(x)$ 와 $B(x)$ 의 자료구조를 linked list로 수정하는 것은 불허)
- 곱하기 결과로 $D(x)$ 의 항들을 임시로 항 배열 등의 배열에 저장했다가 이를 linked list로 전환하는 경우에는 **0점 처리** (즉, $D(x)$ 를 위한 공간으로 linked list 외에 어떤 형태의 배열 및 그런 배열에 준하는 공간의 사용 불허)
- $D(x)$ 의 항들을 지수값으로 sorting 알고리즘을 수행하여 정렬하는 경우에는 **0점 처리** (참고로, 5번 요건에서 $D(x)$ 를 위한 배열 할당을 불허하고 있으므로 sorting 알고리즘의 적용은 사실 불가능함)

아래 사항들은 과제 I 과 동일 (단, 3. 프로그램 구성의 (2)는 수정됨):

- 다항식의 계수 및 지수 값:** 계수 값은 양수 또는 음수이고 자료형은 정수, 지수 값은 0 이상 9 이하로 자료형은 정수

- 다항식 값의 제약:** $A(x)$ 와 $B(x)$ 의 항의 수는 각각 3개로 한다. 단, $A(x)$ 와 $B(x)$ 항들의 지수값은 $D(x)$ 의 항의 수가 9개 미만인 되도록 선택한다.

예시: axb 는 ax^b 를 표시

$$A(x) = 2x^5 + 7x^2 - 4x^0$$

$$B(x) = 6x^3 + 2x^1 + 9x^0$$

$$\begin{aligned} D(x) &= 12x^8 + 4x^6 + (42+18)x^5 + (-24+14)x^3 + 63x^2 - 8x^1 - 36x^0 \\ &= 12x^8 + 4x^6 + 60x^5 - 10x^3 + 63x^2 - 8x^1 - 36x^0 \end{aligned}$$

이 예에서 $D(x)$ 의 x^5 항과 x^3 항은 곱하기 과정에서 각각 2회 계산되어 나오므로 $D(x)$ 의 항 수는 9개가 아닌 7개가 된다.

3. 프로그램 구성:

- $A(x)$ 와 $B(x)$ 를 입력받아 계수 배열 (지수 오름순)에 각각 저장한다.
- $A(x)$ 와 $B(x)$ 를 곱하여 **$D(x)$ 를 linked list에 저장한다.**
- $A(x)$, $B(x)$, $D(x)$ 를 출력한다.
- 이상 1~3의 과정을 $A(x)$ 와 $B(x)$ 다항식을 바꾸어 가며 3회 반복한다.

- $A(x)$, $B(x)$ 의 출력:** 계수와 지수를 입력 받은 값으로 바로 출력하는 것이 아니라 지정된 표현방법으로 저장한 후, 저장된 자료구조로부터 읽어서 출력한다. 즉, 지정된 표현방법으로 정확히 저장되었는지 확인하는 것이 목적이다. 계수가 0인 항도 출력한다.

- $D(x)$ 의 출력:** 지수 내림순으로 출력한다.

프로그래밍 언어: C 언어로 한정

레포트 내용 구성:

1. D(x) 이외의 부분에서 과제 I 프로그램의 수정 사항: 수정 내용 및 사유 설명. 수정이 없는 경우는 '수정 없음'으로 기술

2. 구현한 곱하기 알고리즘에 대한 설명 (해당 부분 C 코드 캡처 이미지를 삽입하고 설명)

특히, A(x)의 1개 항 ax^i 과 B(x)의 1개 항 bx^j 을 곱해서 항 abx^{i+j} 를 산출했을 때, D(x)를 저장하는 linked list에 지수 $=i+j$ 인 노드가

(1) 없고 list가 empty인 경우 (즉, 항 abx^{i+j} 를 표현한 노드가 linked list의 유일한 노드가 되는 경우),

(2) 없고 list가 empty가 아닌 경우,

(3) 이미 있고 항 abx^{i+j} 를 반영하였을 때 계수가 0 이 아닌 경우,

(4) 이미 있고 항 abx^{i+j} 를 반영하였을 때 계수가 0 이 되는 경우

에 각각 어떻게 처리했는지와 자신의 그러한 구현에 장단점이 있다면 무엇인지에 대한 설명

3. 프로그래밍의 어떤 부분에 있어 그것의 구현 방식이 다양한 경우, (1) 어떤 부분이며, (2) 어떤 구현 방식을 선택 했는지, 그리고 (3) 자신의 선택의 이유를 각 경우마다 설명: 과제 I을 예로 들자면, "A(x)와 B(x)의 입력" 부분 등이 구현 방식이 다양한 경우에 해당됨

4. 프로그램 실행결과 화면 캡처 및 설명

5. 가정(assumption) 등 기타사항 (필요시)

제출물:

1. 레포트 파일 (파일 형식은 pdf)

2. 소스코드 파일: .c 또는 .txt 파일로 제출. (소스코드만 조교 컴퓨터의 VS 프로젝트로 복사되어 컴파일/실행 예정)

Visual Studio 버전: 조교의 "프로그램 과제 제출 및 채점 안내" (3월22일자 eClass에 공지)에 따름

C 코드 제출 준수사항: 조교의 "프로그램 과제 제출 및 채점 안내" (3월22일자 eClass에 공지)에 따름

제출처: eClass 과제방

제출마감일시: 5월11일(수) 23시50분

지연제출 감점 규정 및 제출마감 유의사항: "과제 제출마감 유의사항" (3월21일자 eClass에 공지) 필독

기타 사항: 수업시간에 설명한 내용에 따름