

## [고등부] 16회 서울 고등부 본선

### 문제 1. 다항식으로 변형

자판으로부터 괄호로 묶여진 수식을 입력받은 후 전개하여 다항식의 형태로 바꾸어 출력하는 프로그램을 <처리 조건>에 따라서 작성하시오.

<처리 조건>

- (1) 괄호로 묶여진 수식과 이것을 전개한 다항식의 예를 들면 다음과 같다.

$$(X+1)(X-2XY+Y^2) \text{ ----> } X^2+X-2XY+XY^2-2X^2Y+Y^2$$

괄호로 묶여진 수식

전개된 다항식

(입력)

(출력)

- (2) 수식에 사용되는 문자는 영문자(A~Z), 숫자(0~9), 괄호(()), 덧셈(+), 뺄셈(-), 부호(+, -) 이다.
- (3) 전개할 수식이  $(2X+(XY^3-2Z)^2)^{-3}$  이라면 이 식을 컴퓨터에 입력할 때에는 다음과 같은 방법으로 입력한다. 즉, 거듭제곱은 기호 ^를 사용한다.

$$(2X+(XY^3-2Z)^2)^{-3}$$

- (4) 정리되지 않은 항들은 하나의 항으로 정리한다. 예를 들면 다음과 같다.

$$-2XY^2+3XY^2 \text{ ----> } XY^2$$

- (5) <입출력 형식>과 같이 전개된 다항식의 결과는 각 항에 사용된 영문자를 기준으로 오름차순 정렬한다.

<입출력 형식>

수식입력 : $(X+Y)(X^2-(Y+Z^2)^2)$ <span style="border: 1px solid black; padding: 0 2px;">Enter</span>
결과 : $3 \ 2 \ 2 \ 2 \ 4 \ 3 \ 4 \ 2 \ 2$ $X \ +X \ Y \ -XY \ -2XYZ \ -XZ \ -Y \ -YZ \ -2Y \ Z$

### 문제 2. 지도 판별

건설회사 A와 B에서는 새로 지은 건물의 지도를 그리기 위해 두 건설회사에서 한 명씩의 직원을 선발하였다. 두 직원이 그린 지도가 서로 일치하는지 여부를 판별하는 프로그램을 <처리 조건>에 따라서 작성하시오.

<처리조건>

- (1) 건물은 복도와 방으로 구성되어 있다.
- (2) 각 복도의 이름은 영어 소문자 a~z 중 하나의 문자로 붙여져 있다.
- (3) 방은 이름이 없으며 직원이 각 방마다 임의의 숫자로 이름을 붙일 수 있다.
- (4) 한 개의 방에 여러 개의 복도가 연결될 수 있다.
- (5) 지도를 그리는 방법은 다음과 같다. 현재 있는 방에 문이 2개가 있다고 하자. 이 두 개의 문에 이어진 복도는 이름이 각각 r 과 t 이며, 현재의 방에서 r 복도로 연결된 방을 1이라 하고 t 복도로 연결된 방을 2 라면 다음과 같이 연결된 지도를 식으로 나타낸다.

r 1 t 2

- (6) 다음은 두 사람이 만든 지도를 컴퓨터에 입력한 예이다.

```

방수 : 5 [Enter]
A1 : a 2 [Enter]
A2 : t 5 r 4 [Enter]
A3 : a 4 [Enter]
A4 : r 2 t 3 [Enter]
A5 : b 5 i 5 o 5 [Enter]
B1 : a 3 [Enter]
B2 : a 4 [Enter]
B3 : r 4 t 1 [Enter]
B4 : r 3 t 5 [Enter]
B5 : b 5 o 5 i 5 [Enter]
    
```

- ① ‘방수 :’ 다음에는 방의 수를 입력한다. 5를 입력하면 건물 내부에 5개의 방이 있다는 의미이다. 단, 방의 개수는 2부터 10까지로 한다.
  - ② ‘A1 :’에는 A건설회사 직원이 조사한 자료 중 1번 방과 연결된 다른 방과 복도에 대한 식을 입력한다.
  - ③ ‘A1 :’ 다음에 ‘a 2’를 입력한 것은 ‘1번 방’이 ‘a’ 복도를 통해 ‘2번 방’과 연결되었음을 의미한다.
  - ④ ‘방수 :’를 입력한 후 [Enter] 키를 치면 ‘A1 :’이 자동으로 나오게 하고 식 입력 후 [Enter] 키를 치면 ‘A2 :’가 나타난다. 이후 ‘A5 :’까지 같은 방법으로 식을 입력한다.
  - ⑤ B건설회사 직원이 작성한 식도 마찬가지로 ‘B1 :’부터 ‘B5 :’까지 입력한다.
- (7) A건설회사 직원의 지도와 B건설회사 직원의 지도가 일치하면 A건설회사 직원이 지도상에 표시한 방 번호 1, 2, 3, 4, 5와 일치하는 B건설회사 직원이 표시한 방번호를 차례로 출력한다.
  - (8) <입출력 형식>은 다음과 같다.

<입출력 형식>

```

방수 : 5 [Enter]
A1 : a 2 [Enter]
A2 : t 5 r 4 [Enter]
A3 : a 4 [Enter]
A4 : r 2 t 3 [Enter]
A5 : b 5 i 5 o 5 [Enter]
B1 : a 3 [Enter]
B2 : a 4 [Enter]
B3 : r 4 t 1 [Enter]
B4 : r 3 t 5 [Enter]
B5 : b 5 o 5 i 5 [Enter]

A건설회사 직원 : 1 2 3 4 5
B건설회사 직원 : 2 4 1 3 5
    
```

(9) 두 직원이 그린 지도가 일치하지 않을 경우에는 다음과 같이 “일치하지 않음”이라고 출력한다.

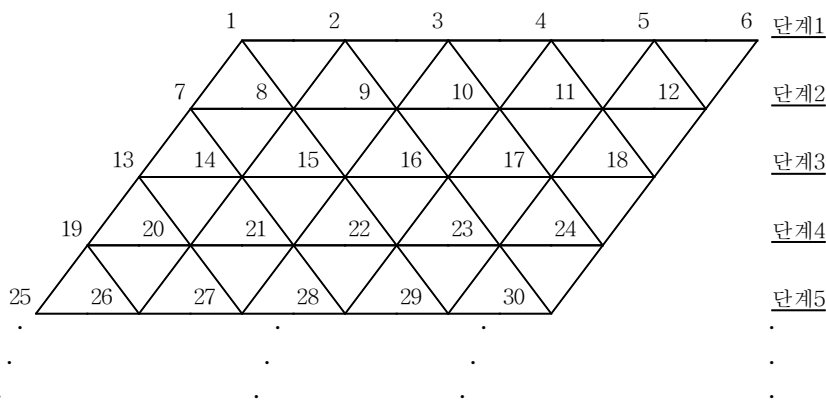
일치하지 않음

## 문제 3. 도형면적계산

자판으로부터 도형의 위치를 숫자로 입력받은 후 입력된 숫자를 차례로 연결하여 형성되는 도형의 모양 및 면적을 계산하는 프로그램을 <처리 조건>에 따라서 작성하시오.

<처리조건>

(1) <그림 3-1>에서 단계는 최대 10까지로 한다.



<그림 3-1 >

단계10

(2) 도형의 위치는 자판으로부터 입력받는다. 도형의 위치를 알아보고자 할 때 입력되는 숫자의 개수는 3 개부터 10개까지이다.

도형의 위치? 1 13 15 Enter

(3) 도형의 형태는 자판으로부터 입력받은 숫자를 차례로 연결했을 때 실선으로 형성되는 것으로 삼각형, 평행사변형, 마름모, 육각형이다.

(4) 입력 범위(1부터 60까지)를 벗어나면 다음과 같은 오류 메시지를 출력한 후 다시 입력받도록 한다.

도형의 위치? 1 61 3 Enter  
 >> 입력 범위 오류

(5) 삼각형, 평행사변형, 마름모, 육각형 이외에 발생하는 도형은 다음과 같은 메시지를 출력한 후 다시 입력받도록 한다.

도형의 위치? 3 15 22 Enter  
 >> 도형의 위치 오류

(6) 자판으로부터 입력받은 숫자를 차례로 연결한 뒤, 실선으로 형성되는 도형을 판단할 때 시작 지점에 상관없이 입력된 숫자를 차례로 연결하여 도형이 성립되면 된다. 예를 들어 다음과 같이 입력되면 모

두 삼각형으로 취급한다.

```

도형의 위치? 1 7 8 [Enter]
도형 => 삼각형
면적 => 5
도형의 위치? 7 8 1 [Enter]
도형 => 삼각형
면적 => 5
도형의 위치? 8 1 7 [Enter]
도형 => 삼각형
면적 => 5
    
```

(7) <그림 3-1>에서 선택한 도형의 면적을 계산하는 방법은 다음 표와 같다. 기본적으로 <그림 3-1>에서 실선으로 그려진 가장 작은 삼각형 1개의 면적은 5 이다.

도형의 모양	도형의 위치	도형의 면적
삼각형	1 7 8	5
삼각형	1 13 15	20
평행사변형	1 13 14 2	20
마름모	7 19 33 21	40
육각형	3 9 16 17 11 4	30

(8) <입출력 형식>은 다음과 같다.

① 예제 1

```

도형의 위치? 7 8 1 [Enter]
도형 => 삼각형
면적 => 5
    
```

② 예제 2

```

도형의 위치? 2 14 28 16 [Enter]
도형 => 마름모
면적 => 40
    
```

## 문제 4. 정사각형 구성

한 변이 1[cm]인 정사각형으로 구성된 “평면블록”을 조합하여 한 변이 3[cm]인 정사각형을 구성할 수 있는지 여부를 판별하는 프로그램을 <처리 조건>에 따라서 작성하시오.

<처리조건>

- (1) 한 변이 1[cm]인 정사각형으로 구성된 평면블록의 종류는 <그림 4-1>과 같이 8가지로 주어진다.

종류	A형	B형	C형	D형
모양	□ 1cm×1cm	□□	□□□	□□ □
종류	E형	F형	G형	H형
모양	□□□□ □	□□□ □□	□□□ □□	□□ □□

< 그림 4-1 >

(2) 프로그램을 실행하면 <입력 형식>과 같이 조합할 블록의 종류를 영문자로 입력받는다.

- ① 같은 모양의 블록을 2개 이상 입력할 수 있다.
- ② 입력한 각 블록을 구별하기 위하여 “입력 순번”을 각 블록의 고유번호로 사용한다.
- ③ 입력되는 블록의 총 개수는 9개 이하이다.
- ④ ‘0’을 입력하면 입력의 종료를 의미한다.

<입력 형식>

```
1번째 블록 ? G [Enter]
2번째 블록 ? B [Enter]
3번째 블록 ? B [Enter]
4번째 블록 ? A [Enter]
5번째 블록 ? 0 [Enter] ('0'은 블록입력의 종료를 의미함)
```

(3) 위에서 입력한 블록을 모두 사용하여 한 변이 3[cm]인 정사각형을 구성할 수 있는지 여부를 판별하여 출력한다. 출력할 내용은 블록을 구별하기 위해 사용한 블록의 “입력 순번”이며, 다음 규칙을 사용하여 판별한다.

- ① 입력한 블록은 모두 한번씩만 사용해야 한다.(블록의 종류를 A형만 4번 입력하였다면 A블록을 4번 사용해야 한다.)
- ② 각 블록에 대하여 회전, 이동 또는 뒤집을 수 있다.
- ③ 숫자의 배열이 좌, 우 또는 상, 하로 위치만 바뀌었더라도 별 개의 조합으로 간주한다.
- ④ 출력 순서는 무관하며, 출력할 자료가 5개 이상일 경우 1줄에 5개 씩 출력한다.

(4) <입력 형식>에 대한 <출력 형식>은 다음과 같다. 단, <출력 형식>에서 선은 출력하지 않는다.

<출력 형식>

<table border="1" style="display: inline-table; margin-right: 20px;"> <tr><td>2</td><td>2</td><td>3</td></tr> <tr><td>4</td><td>1</td><td>3</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table> <table border="1" style="display: inline-table; margin-right: 20px;"> <tr><td>3</td><td>2</td><td>2</td></tr> <tr><td>3</td><td>1</td><td>4</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table> <table border="1" style="display: inline-table; margin-right: 20px;"> <tr><td>2</td><td>3</td><td>3</td></tr> <tr><td>2</td><td>1</td><td>4</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table> <table border="1" style="display: inline-table; margin-right: 20px;"> <tr><td>2</td><td>4</td><td>3</td></tr> <tr><td>2</td><td>1</td><td>3</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table> <table border="1" style="display: inline-table;"> <tr><td>3</td><td>3</td><td>2</td></tr> <tr><td>4</td><td>1</td><td>2</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	2	2	3	4	1	3	1	1	1	3	2	2	3	1	4	1	1	1	2	3	3	2	1	4	1	1	1	2	4	3	2	1	3	1	1	1	3	3	2	4	1	2	1	1	1
2	2	3																																											
4	1	3																																											
1	1	1																																											
3	2	2																																											
3	1	4																																											
1	1	1																																											
2	3	3																																											
2	1	4																																											
1	1	1																																											
2	4	3																																											
2	1	3																																											
1	1	1																																											
3	3	2																																											
4	1	2																																											
1	1	1																																											
.....(이하 생략)																																													

총 ( 24 )가지

(5) 조건을 만족하는 숫자(블록순번)의 조합이 전혀 없으면 “구성할 수 없음”을 출력하며, 반면에 조건을 만족하는 경우가 너무 많아 한 화면을 벗어 날 경우는 [스페이스바] 키를 누를 때마다 다음 화면이 출력되도록 한다.

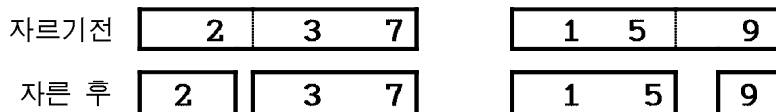
(6) 더 이상 출력할 자료가 없을 경우에는 <출력 형식>과 같이 출력된 총 가지수를 출력한다.

## 문제 5. 카드숫자 조합

한 장의 카드 앞면과 뒷면에 각각 같은 자릿수의 임의의 숫자가 있다. 이 카드를 아래 그림과 같이 자르면 두 장의 카드조각으로 나뉘어진다. 그러면 각기 다른 4개의 숫자가 만들어진다.(종이를 자르는 방법은 이외에도 여러 가지로 나눌 수 있다.) 이렇게 하여 만들어진 여러 개의 숫자들을 조합하여 더했을 때, 더한 값이 특정한 수에 가장 근접한 값을 찾는 프로그램을 <처리 조건>에 따라서 작성하시오.

( 앞면 )

( 뒷면 )



<처리조건>

- (1) 카드의 앞면에 237을 입력하고 뒷면에 159를 입력하되, 앞면에 2, 3, 7은 뒷면의 9, 5, 1과 차례로 일치하도록 한다.
- (2) 앞면에 237을 표시하고 뒷면에 9를 입력한다면 9의 앞에 0을 두 개 입력하여 009로 만들어 앞면과 뒷면의 자릿수를 일치하도록 한다.
- (3) 카드에 표시할 수 있는 숫자는 양의 정수 1000000000000 미만으로 한다.
- (4) 카드의 양면에 237/159(앞면/뒷면)를 입력한 뒤, 카드를 자르는 방법은 다음과 같이 3가지가 있다.
  - ① 2와 37사이를 잘라내는 방법. 각 조각의 숫자는 2/9(앞면/뒷면), 37/15(앞면/뒷면) 이다.
  - ② 23과 7사이를 잘라내는 방법. 각 조각의 숫자는 23/59(앞면/뒷면), 7/1(앞면/뒷면) 이다.
  - ③ 2와 3, 다시 3과 7사이를 잘라내어 3조각으로 만드는 방법. 이때 각 조각의 숫자는 2/9, 3/5, 7/1 이다.
- (5) 이렇게 잘라낸 카드조각들에 표시된 숫자를 더하는 방법은 다음과 같다.
  - ① 조각 앞뒤의 숫자를 더할 수 있다. 어떤 조각의 숫자가 2/9(앞면/뒷면)이라면 2와 9를 더 할 수 있다.
  - ② 2개 이상의 조각들을 선택하여 해당 조각들의 앞면 또는 뒷면의 숫자들을 선택하여 더 할 수 있다. 2/9(앞면/뒷면)와 37/15(앞면/뒷면)를 선택하였다면 2+9, 2+9+37, 37+15, 2+15+37, 9+37, ... 등, 같은 면의 수를 중복하지 않으면 어느 숫자와도 더할 수 있다.
- (6) 만약 원하는 목표숫자가 K이고 카드를 잘라내서 더한 숫자를 N 이라고 할 때 K를 넘지 않는 가장 근접한 숫자  $N(N \leq K)$ 을 구한다..
- (7) 입력자료는 자판으로부터 다음과 같은 형태로 입력받는다.

입력자료 : 534 197 50 [Enter]

- ① 첫 번째 숫자는 카드의 앞면에 표시할 숫자이다.
  - ② 두 번째 숫자는 카드의 뒷면에 표시할 숫자이다.
  - ③ 마지막 숫자는 찾고자 하는 목표숫자이다.
- (8) <입출력 형식>은 다음과 같다.

① 예제 1

입력자료 : 534 197 50 [Enter]  
 결과 : 41

② 예제 2

입력자료 : 671315 311413 70 [Enter]  
 결과 : 70