

## Creaturi 2: Strigoi



Ana și Ecaterina se plăcătesc. În cei 400 de ani de când bântuie *Padurea Codrilor Întunecatî*, singura lor distracție sunt rarei călători rătăciți. Așadar, ele s-au apucat de informatică.

După ce au învățat de concepte precum *Object Oriented Programming* și *Data Oriented Programming*, cele două au inventat paradigma de *Comment Oriented Programming*. Un program se numește *comment-complet* dacă poate evalua orice funcție prin comentarea (ștergerea) unor linii de cod.

Pentru a testa noul concept, cele două vor să creeze un program *comment-complet* care să evaluateze polinoame. Își propun să rezolve următorul scenariu:

1. Ana alege un număr  $N$ , pe care îl spune Ecaterinei.
2. Ecaterina îi spune Anei un program *comment-complet* pentru polinoamele de grad  $N$ .
3. Ana alege un polinom  $P$  de grad  $N$ , cu coeficienții numere naturale, pe care îl spune Ecaterinei.
4. Ecaterina îi spune Anei care linii din programul ei trebuie comentate, pentru ca programul obținut să calculeze  $P(X)$ .

Instrucțiunile permise în program sunt:

- **\$VAR1 = \$VAR2 [+\*] \$VAR3** – Valoarea variabilei **\$VAR1** devine suma / diferența / produsul variabilelor **\$VAR2** și **\$VAR3**;
- **\$VAR1 = \$VAR2 [+\*] A** – Valoarea variabilei **\$VAR1** devine suma / diferența / produsul variabilei **\$VAR2** și numărul **natural A**;
- **\$VAR1 = A** – Valoarea variabilei **\$VAR1** devine numărul **natural A**;

Variabilele pot avea orice nume format din litere, cifre și underscore ("\_"), și există două variabile speciale:

- **\$X**: Variabila reprezintă valoarea în care dorim să evaluăm polinomul  $P$ .
- **\$Y**: După executarea programului, această variabilă trebuie să fie egală cu  $P(\$X)$ .

## Interacțiune

Întâi citiți gradul  $N$  al polinomului.

După aceea, afișați pe o linie  $L$ , reprezentând numărul de instrucțiuni, urmat de  $L$  linii, programul *comment-complet*.

După ce ați afișat programul (**doar după**), veți putea citi valorile  $P_0, P_1, \dots, P_N$ , coeficienții polinomului  $P$ .

După ce ați citit polinomul  $P$ , afișați  $C$ , numărul de linii de comentat, urmat de  $C$  valori, indicii linilor (indexate de la 0).

**Atenție:** Nu uitați să dați flush la `stdout` cu `cout.flush()` (C++) sau `fflush(stdout)` (C).

## Constrângeri

- $0 \leq N \leq 10^3$ .
- $0 \leq P_i \leq 10^9 \quad \forall 0 \leq i \leq N$  (**Important**).
- Programul generat poate avea cel mult  $2 * 10^5$  operații.

## Subtask-uri

1. (20 de puncte)  $N = 1$  și  $0 \leq P_i \leq 100 \quad \forall 0 \leq i \leq N$ .
2. (20 de puncte)  $0 \leq N \leq 100$  și  $0 \leq P_i \leq 100 \quad \forall 0 \leq i \leq N$ .
3. (30 de puncte)  $N = 0$ .
4. (30 de puncte) Nicio constrângere suplimentară.

## Exemplu

**Notă:** Programul afișat în acest exemplu este *comment-complet* numai pentru polinoame cu coeficienții cel mult 2.

Input Standard ( <i>cin</i> )	Output Standard ( <i>cout</i> )
1	<pre> 8 \$Y0 = 0 \$Y0 = 1 \$Y0 = 2 \$Y1 = 0 \$Y1 = 1 \$Y1 = 2 \$Y1 = \$Y1 * \$X \$Y = \$Y0 + \$Y1 </pre>
1 2	<pre> 4 0 2 3 4 </pre>
0 1	<pre> 8 \$Y0 = 0 \$Y0 = 1 \$Y0 = 2 \$Y1 = 0 \$Y1 = 1 \$Y1 = 2 \$Y1 = \$Y1 * \$X \$Y = \$Y0 + \$Y1 </pre>
	<pre> 4 1 2 3 5 </pre>

## Explicație Exemplu

Considerăm primul exemplu.

### Pas 1

Ana a ales un polinom de grad 1. Așadar, îi transmite Ecaterinei numărul 1.

### Pas 2

Ecaterina răspunde cu următorul program:

```

$Y0 = 0
$Y0 = 1
$Y0 = 2
$Y1 = 0
$Y1 = 1
$Y1 = 2
$Y1 = $Y1 * $X
$Y = $Y0 + $Y1

```

### **Pas 3**

Ana îi spune Ecaterinei că a ales polinomul  $P(X) = 1 + 2 * X$ .

### **Pas 4**

Ecaterina îi spune Anei că pentru ca programul pe care l-a afișat la pasul 2 să evalueze polinomul P, trebuie comentate liniile 0, 2, 3 și 4.

### **Verificare**

După comentarea liniilor, codul Ecaterinei devine:

```
$Y0 = 1  
$Y1 = 2  
$Y1 = $Y1 * $X  
$Y = $Y0 + $Y1
```

Putem să observăm aşadar că  $\$Y = 1 + 2 * \$X = P(\$X)$ .