Zadanie **Dostęp**

Limit czasu: 0.2s

OLIMPIJSKIE KOŁO INFORMATYCZNE

Treść

Systemy typu Unix/Linux mają bardzo ciekawe podejście do zarządzania **dostępem** użytkownika do danego pliku. Zapisują tę informację w postaci jednej liczby całkowitej, z której później mogą odczytać dane uprawnienie dla wyznaczonej grupy uprawnień (są trzy - właściciel pliku, przypisana grupa użytkowników i cała reszta użytkowników niezawarta w powyższych grupach).

Działa to w sposób następujący: aby odczytać uprawnienia dowolnego użytkownika (grupa other), system rozważa 3 ostatnie bity zapisu binarnego danej liczby, z czego każdy odpowiada za jedno z podstawowych uprawnień pliku/folderu.

		r	w	x	r	w	x	r	w	x
·	·	·	1	1	0	0	1	0	1	0	1
				właściciel			grupa			other

Poszczególne bity odpowiadają uprawnieniom: read (pozwala wyświetlać zawartość pliku) lub folderu, write (pozwala modyfikować lub usuwać jego zawartość) oraz execute (pozwala wykonywać program). Jeśli bit odpowiadający danej czynności ma wartość 1, użytkownik może wykonać daną czynność, a jeśli 0, to nie.

Wejście

W pierwszej i jedynej linii wejścia znajdują się jedna liczba całkowita n ($-10^{18} \leq n \leq 10^{18}$) będąca zapisem uprawnień użytkownika w systemie, który opisano powyżej.

Wyjście

Wyjście powinno składać się z trzech wierszy zawierających odpowiedź na to, czy dany użytkownik ma uprawnienia do danych czynności - odpowiedno **read**, **write** i **execute**.

Przykłady

Wejście (przykład 1):

```
44
```

Wyjście (przykład 1):

```
TAK
NIE
NIE
```

Wyjaśnienie (przykład 1):

Liczba 44_{10} w zapisie dwójkowym to 101100_2, czyli ostatnie bity to 1 0 0.
- Bit na indeksie 0 wynosi 0, czyli nie można wykonać pliku.
- Bit na indeksie 1 wynosi 0, czyli nie można modyfikować pliku.
- Bit na indeksie 2 wynosi 1, czyli można odczytać plik.

Autor: Adam Duda