Zadanie Kod binarny Fibonacciego – LOGIA 22 (2021/22), etap 3

Treść zadania

W systemie binarnym do zapisu liczb wykorzystywane są tylko dwie cyfry: 0 i 1. Aby obliczyć wartość dziesiętną liczby zapisanej w naturalnym kodzie binarnym wystarczy wymnożyć jej cyfry przez odpowiednie potęgi liczby 2 i dodać wyniki.

Na przykład: $10100_2 = 1*2^4 + 0*2^3 + 1*2^2 + 0*2^1 + 0*2^0 = 16 + 4 = 20$

Tosia postanowiła zamiast potęg liczby 2 wykorzystać liczby Fibonacciego. Kolejna liczba Fibonacciego jest sumą dwóch poprzednich. Przyjęła za dwie pierwsze liczby 1 i 2. Otrzymała w ten sposób liczby Fibonacciego 1, 2, 3, 5, 8, 13, 21, …

Na przykład: $10100_F = 1*8 + 0*5 + 1*3 + 0*2 + 0*1 = 8 + 3 = 11$

Tosia zrobiła ponadto założenie, że w kodzie binarnym Fibonacciego dwie jedynki nie mogą wystąpić obok siebie, ponieważ zapis nie byłby jednoznaczny (np. $11_F = 1*2 + 1*1 = 3$,

$100_F = 1*3 + 0*2 + 0*1 = 3$).

Poniższa tabela przedstawia zapis kolejnych liczb w kodzie binarnym Fibonacciego.

Liczba dziesiętna	Kod binarny Fibonacciego
1	1
2	10
3	100
4	101
5	1000
6	1001
7	1010
8	10000
9	10001
10	10010

Pomóż Tosi i napisz program, który zamienia liczbę dziesiętną na kod binarny Fibonacciego.

Wejście:

Liczba naturalna składająca się maksymalnie z 1000 cyfr.

Wyjście:

Zapis liczby z wejścia w kodzie binarnym Fibonacciego.

Ośrodek Edukacji Informatycznej
i Zastosowań Komputerów w Warszawie

	Przykład 1	*Przykład 2*	*Przykład 3*
Wejście	**3**	**11**	**16**
Wyjście	**100**	**10100**	**100100**

Omówienie rozwiązania

Główny problem w zadaniu to generowanie liczb Fibonacciego. Można zastosować jedno z dwóch podejść: rekurencyjne lub iteracyjne. Rekurencyjne generowanie liczb Fibonacciego jest obarczone ograniczeniami. Jesteśmy w stanie w skończonym czasie uzyskać jedynie niewielkie liczby (do 10 cyfr). W zadaniu daną może stanowić liczba naturalna składająca się z maksymalnie 1000 cyfr. Jeśli nasze zadanie ma uporać się z tak dużymi liczbami musimy postawić na rozwiązanie iteracyjne.

```
lista[0] ← 1
lista[1] ← 2
i ← 1
dopóki lista[i - 1] + lista[i] <= liczba wykonuj
    i ← i + 1
    lista[i] ← lista[i - 2] + lista[i - 1]
```

Ostatnia wygenerowana liczba Fibonnaciego jest nie większa niż liczba, którą poddajemy przekształcaniu.

Liczbę na kod binarny Fibonacciego będziemy zamieniać metodą zachłanną. Przeglądanie liczb Fibonacciego rozpoczynamy od największej wygenerowanej liczby, czyli ostatniego elementu listy. Odpowiada ona najbardziej znaczącej jedynce w binarnym kodzie Fibonacciego. Do dalszego procedowania bierzemy różnicę przekształcanej liczby i liczby Fibonacciego. W kolejnych krokach szukamy pierwszej liczby Fibonacciego równej lub mniejszej aktualnej wartości przekształcanej liczby. Pominięte liczby Fibonacciego odpowiadają zerom w kodzie. W momencie, gdy różnica osiągnie wartość zero, należy wypisać wszystkie zera, które odpowiadają pozostałym liczbom Fibonacciego.

```
dopóki liczba > 0 wykonuj
    jeśli lista[i] <= liczba:
        wypisz 1
        liczba ← liczba - lista[i]
    w przeciwnym przypadku
            wypisz 0
    i ←  i - 1
dopóki i >= 0 wykonaj
    wypisz 0
    i ← i - 1
```

Rozwiązanie w języku Python

```python
def naFib(liczba):
    if liczba < 2:
        print(liczba)
        return
    lista = [1, 2]
    i = 1
    while lista[i - 1] + lista[i] <= liczba:
        i = i + 1
        lista.append(lista[i - 2] + lista[i - 1])
    while liczba > 0:
        if lista[i] <= liczba:
            print('1', end = '')
            liczba = liczba - lista[i]
        else:
            print('0', end = '')
        i = i - 1
    while i >= 0:
        print('0', end = '')
        i = i - 1

liczba = int(input())
naFib(liczba)
```

Uwaga: liczby naturalne mniejsze od 2 wymagają odrębnego rozpatrzenia.

Testy

Pierwsza, druga i trzecia grupa testów uwzględnia rekurencyjne generowanie liczb Fibonacciego. W pierwszej grupie testów żadna z liczb nie jest liczbą Fibonacciego. W drugiej grupie testów są wyłącznie liczby Fibonacciego. Czwarta i piąta grupa testów wymaga iteracyjnego generowania liczb Fibonacciego. Każda kolejna grupa testów zawiera liczby o narastającej liczbie cyfr. Są wśród nich liczby Fibonacciego.

Grupa testów	Liczba Fibonacciego	Test	Wynik
I	nie	12	10101
	nie	31	1010010
	nie	50	10100100
II	tak	13	100000
	tak	55	100000000
	tak	89	1000000000
III	tak	46368	1000000000000000000000
	nie	317812	1000000000000000000000000001
	nie	1000000	1000101000000000000101000000000
IV	nie	1836311904	1000000000000000000000000000000000000000000001
	nie	259695496911122584	1010101010101010101010101010101010101010101010101 0101010101010101010101010101010101010101
	nie	1000000000000000000	1010001010101000100101000010100000000010101010 0000001000100100101001000101001010100000 10

V	tak	testy poniżej	
	nie	testy poniżej	
	nie	testy poniżej	

Test

```
7334343300431526745413249208471480941295886039383885689591799012660621316779395823
7053069359544403744909833164405109382186615030308616305515481762932224684304758
2625995980060421290142898205568182915095065519928778385701506122422679548247462336
7665222282256870316117093830732812895097667473105049114270817452107418279883991688
4500061707705025660630325640753106324747344254906891906352595497679400999050875
0132092929663097227960341493099744427909403937406527274085352003978578669822758831
6593293922583225068083713333591230083257890903615803904675715141475535706043142298
8866792045298371160569065663936383352746627653451155462689725430610815806257367133
6560404013227178632840763397427562452038558146631134906898622106969008675310842657
2531779645697971487832866994680744372888779783943546341379591294049731675861116
8447311213734925988635812151655063724137755355835867847379508102339675238232695173
8968232340614252382608525081448567143195223074804823219291936147167373431494717263
3335223055391230827952220 23 ->
10000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000
00
```

000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000

Test

859050624732544985326232904500976502380626726805069171248498568971528243961342559
196637958677963090329071379033624767494763864292387638110072256994991358314398436
895349075167738480466518675862268037690874335778147151109144101565038899558139648
924882820130330412557472228213892919237913223356577504245797517694893926514556242
981995658727069572421241681585857992483299282790739854631737208539297085697543687
526902189388354690581043351632478969423531308068219234955793971844656883804113959
004299438060221319234900320926018529784711414060427459569525567951779318630722066
951309165044303675474411533749587978897773791063450310802392631903479946042680395
889941572033735123892221830584305736146777747111147632421916483679531703858258081
108288759239503097698195980311121216150 ->
101010101010101010101010101010101010101010101010101010101010101010101010101010101
010101010101010101010101010101010101010101010101010101010101010101010101010101010
101010101010101010101010101010101010101010101010101010101010101010101010101010101
010101010101010101010101010101010101010101010101010101010101010101010101010101010
101010101010101010101010101010101010101010101010101010101010101010101010101010101
010101010101010101010101010101010101010101010101010101010101010101010101010101010

1010101010101010101010101010101010101010101010101010101010101010101010101010101
0101010101010101010101010101010101010101010101010101010101010101010101010101010

999999999999999999999999999999999999999999999999999999999999999999999999999999
999999999999999999999999999999999999999999999999999999999999999999999999999999
999999999999999999999999999999999999999999999999999999999999999999999999999999

0100101010101001010000001010000000010000001001000000101010101000000001000010010000
0000100000100000000101001000100100000010101001000100010101010001010010001000010001
0001000000010101000100101010001000010010010001000000100101010010100101000100010100
1010001001000100100010000100000010100100001010100001000000010000000001010101000100
1010001000000010100001010010000000010000000000010010100000000010001000100100010001000100
0100010000000010000100000100001001000000000010010100010100000000010001000100000100
0100100101000001000000000101000001010010101000010101000101010100100101000000001000
0010100001000010010010000000010101010010000010101000010100001000010010100000100010
0010101010100000010100000000010001000101000100000001010000000001010000101010000010001
0010001000100000001001010101000001010100010101000000010101001001010000101010000010101
0100001000000000100101000101000101001001010001000010001010010100010001001000000100
1010010001010100100100100010100000000101010100001000001000001001010100001010001010
0010100010010001010100100001001000000100010100000100100010000010010010101010101000
0010000000010001000100100010000000100100100100010100101010000000000000000100010100010
1001000100001010100000100000101010000000100100001001000100010101000010100001010010
0000000101010100001010000100000100100000010010101010100001000010001001001000010010010
0010001010010010010010010010000100100010000101000100100000100100100100100101010010000
1010010000101010000010101010001000100000000010010100000000000000000100101010000101010
000101