Matematyka w informatyce

Ciągi określone rekurencyjnie

Zadanie 1:
Oblicz n=3 wyraz ciągu opisanego rekurencyjnie jako:

$$\begin{cases} f_0 = 2 \\ f_n = 3f_{n-1} - 1 \end{cases}$$

Rozwiązanie:

$$f_0 = 2$$
$$f_1 = 3\ f_{1-1} - 1 = 3\ f_0 - 1 = 3 \cdot 2 - 1 = 5$$
$$f_2 = 3\ f_{2-1} - 1 = 3\ f_1 - 1 = 3 \cdot 5 - 1 = 14$$
$$f_3 = 3\ f_{3-1} - 1 = 3\ f_2 - 1 = 3 \cdot 14 - 1 = \mathbf{41}$$

Objaśnienie:
f_n - symbol „n” oznacza numer kolejnego przebiegu rekurencji;
Każdy kolejny przebieg korzysta z wyniku uzyskanego w poprzednim przebiegu.

Zadanie 2:
Oblicz n=2 wyraz ciągu opisanego rekurencyjnie jako:

$$\begin{cases} f_0 = 1 \\ f_n = 5f_{n-1} - 2 \end{cases}$$

Rozwiązanie:

$$f_0 = 1$$
$$f_1 = 5f_{1-1} - 2 = 5f_0 - 2 = 5 \cdot 1 - 2 = 3$$
$$f_2 = 5f_{2-1} - 2 = 5f_1 - 2 = 5 \cdot 3 - 2 = \mathbf{13}$$
$$f_3 = 5f_{3-1} - 2 = 5f_2 - 2 = 5 \cdot 13 - 2 = 63$$

Złożoność obliczeniowa algorytmów

Złożoność to ilość iteracji algorytmu. Istnieje kilka notacji złożoności:
O - standardowa notacja;
Ω - podana jest minimalna ilość czasu, jaka potrzebna jest na wykonanie algorytmu;
Θ - podana jest minimalna i maksymalna ilość czasu potrzebna na wykonanie algorytmu;

Przykłady złożoności w notacji „O”:
$O_{(1)}$ - proste i podstawowe operacje matematyczne, przypisania, odkładanie na stosie, usuwanie ze stosu, itp.

$O_{(n)}$ - wyszukiwanie elementów, pętla FOR (czyli zależy od ilości wykonanych pętli), sortowanie bąbelkowe (w wersji optymistycznej - zależy od początkowego uporządkowania), sortowanie przez wstawianie (w wersji optymistycznej);

$O_{(n^2)}$ - pętla FOR wewnątrz innej pętli FOR, sortowanie bąbelkowe (w wersji pesymistycznej), sortowanie przez wstawianie (w wersji pesymistycznej), sortowanie przez scalanie, sortowanie przez wybór (*selection sort*), sortowanie szybkie (*quick sort*);

$O_{(n^3)}$ - szybkie potęgowanie;

$O_{(n \log(n))}$ - sortowanie przez scalanie, sortowanie szybkie;

Legenda:

n - ilość sprawdzeń / ilość operacji;

Największy wspólny dzielnik

Wzór: **NWD(liczba1, liczba2)** (używa się także angielskiego GCD - *Greatest Common Divisor*)

Algorytm ogólny: Od większej liczby odejmujemy mniejszą tak długo aż obie staną się równe.

NWD(**45**, **27**) = NWD(**45**-27, **27**) =
NWD(**18**, **27**) = NWD(**18**, **27**-18) =
NWD(**18**, **9**) = NWD(**18**-9, **9**) =
NWD(9, 9)

Odpowiedź: Największym wspólnym dzielnikiem jest 9.

NWD(**78**, **13**) = NWD(**78**-13, **13**) =
NWD(**65**, **13**) = NWD(**65**-13, **13**) =
NWD(**52**, **13**) = NWD(**52**-13, **13**) =
NWD(**39**, **13**) = NWD(**39**-13, **13**) =
NWD(**26**, **13**) = NWD(**26**-13, **13**) =
NWD(13,13)

Odpowiedź: Największym wspólnym dzielnikiem jest 13.

Algorytm Euklidesa:

nwd(17,3)

trzy możliwości:

albo **a=b**

albo **a>b** to **a <-- a - b**

albo **a<b** to **b <-- b - a**

Kongruencja

Oznacza równoważność, przystawanie, zgadzanie się (łac. *congruere*). Kongruentne są na przykład liczby, które mają tę samą **resztę** (*modulo*) z dzielenia przez „*n*”. Są one zawsze wielokrotnością „*n*”.

a ≡ b (mod n), np.

57 ≡ 9 (mod 24), gdyż:

57 mod 24 = 9

9 mod 24 = 9

Liczby pierwsze

Duża liczba pierwsza musi być liczbą nieparzystą.

Triangulacja

Podział złożonych wielokątów na trójkąty i obliczanie pola powierzchni tych trójkątów. Jeśli jest to niemożliwe, wtedy wielokąt dzielimy na mniejsze wielokąty, a te dopiero na trójkąty.

Ciąg Fibonacciego

Każda liczba w tym ciągu jest sumą dwóch poprzednich liczb:

F_0 = 0 (umowa w matematyce formalnej)

F_1 = **1** (założenie wstępne)

F_2 = **1** (bo 0+1)

F_3 = **2** (bo 1+1)

F_4 = **3** (bo 1+2)

F_5 = **5** (bo 2+3)

F_6 = **8** (bo 3+5)

F_7 = **13** (bo 5+8)

F_8 = **21** (bo 8+13)

itd.

Zbiory i permutacje

Permutacja to wszystkie możliwości rozpisania kodu. Liczba permutacji **zbioru** składającego się z 4 elementów to **4!** (czyli 24), składającego się z 5 elementów to **5!** (czyli 120).

W przypadku multizbioru (elementy mogą się powtarzać, np. {1,2,2,3}) - będzie to **4!/1! * 2! * 1!**

Jeżeli element podzbioru **a={1,3,5,6,7}** należy do zbioru **x={1,2,4,5,8}** to 10100 (bo tylko jedynka podzbioru "a" należy do zbioru "x").

Permutacja to ciąg wszystkich elementów **multizbioru** (istotna jest kolejność wyrazów), np. ze zbioru **{1,2,2,4}**, permutacją jest: **(1,2,2,4) (2,2,1,4)** itd.

W przypadku liter i symboli, przypisujemy im liczby całkowite.

Czy liczba jest podzielna przez 3 lub 9?

Sposób nie-wprost:

12345 : **3** = (1+2+3+4+5) : **3** = 15 : 3 = **5** (więc liczba 12345 także jest podzielna przez 3)

12345 : **9** = (1+2+3+4+5) : **9** = 15 : 9 = **1,66** (więc liczba 12345 też nie jest podzielna przez 9)

Inne przykłady:

1 046 256 : **3** = (1+0+4+6+2+5+6) : **3** = 24 : 3 = **8** (więc 1046256 też jest podzielne przez 3)

7 607 988 : **9** = (7+6+0+7+9+8+8) : **9** = 45 :9 = **5** (więc 7607988 też jest podzielne przez 9)

13 332 : **3** = (1+3+3+3+2) : **3** = 12 : 3 = **4** (więc 13332 też jest podzielne przez 3)

59 914 188 : **9** = (5+9+9+1+4+1+8+8) : **9** = 45 : 9 = **5** (więc 59914188 też jest podzielne przez 9)

28 551 : **3** = (2+8+5+5+1) : **3** = 21 : 3 = **7** (więc 28551 też jest podzielne przez 3)

28 551 : **9** = (2+8+5+5+1) : **9** = 21 : 9 = **2,33** (więc 28551 też nie jest podzielne przez 9)

Uwaga: Co ciekawe, w badaniu podzielności przez 3 możemy pominąć sumowanie takich liczb jak 0,3,6 9.

Podzielność przez 7

Dzielimy dużą liczbę na 6-cyfrowe grupy (zaczynając od tyłu). Jeśli suma tych grup dzieli się przez 7, cała liczba także dzieli się przez 7. Na przykład:

000002 563662 938488 212234 238743 643746 873622 =

000002 + 563662 + 938488 + 212234 + 238743 + 643746 + 873622 = 3470497 % 7 = 7

Odpowiedź: Liczba jako całość nie dzieli się przez 7.

Podzielność przez 4

Gdy dwie ostatnie cyfry dzielą się przez 4.

Podzielność przez 5

Gdy liczba kończy się na 0 lub na 5.

Ostatnia aktualizacja: 30 października 2023.