

Zadanie 10

Szereg Fibonacciego zadany jest wzorem

$$F_n = \begin{cases} 0 & n = 0 \\ 1 & n = 1 \\ F_{n-1} + F_{n-2} & n > 1 \end{cases} \quad (1)$$

Powyższą relację można zapisać także jako

$$\begin{pmatrix} F_{k+2} \\ F_{k+1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} F_{k+1} \\ F_k \end{pmatrix} \quad (2)$$

$$F_0 = 0 \quad F_1 = 1 \quad (3)$$

Proszę zaimplementować funkcje, liczące n -ty wyraz ciągu Fibonacciego następującymi metodami:

1. Funkcyjnie (należy napisać *funkcję* `fibFunc` taką, by `Head[fibFunc]` => `Function`, można użyć n.p. instrukcji `Switch`),
2. Iteracyjnie (należy napisać funkcję lub regułę, która wykorzystuje pętlę `Do` iterując relację(2))
3. Przy użyciu reguł (należy symbolowi `fibRule` przypisać odpowiednie reguły korzystając z konstrukcji `/;`),
4. Używając programowania dynamicznego i zapisując wygenerowane reguły w symbolu `fibDyn` (nieobowiązkowe).
5. Używając wektorów własnych macierzy z (2) zapisać w jawny sposób wzór na F_n (nieobowiązkowe).

Proszę użyć polecenia `Timing` dla `fibRule`, `fibFunc` oraz `fibDyn` by sprawdzić czas obliczenia F_{25} a następnie narysować wykresy czasu wykonywania każdej metody w zależności od n dla maksymalnego n dobrane do metody.

1 Zadanie 11

- Proszę napisać własną implementację funkcji `Length` używając wzorców.
- Proszę zaimplementować funkcję `part[list,n]`, która zwraca n -ty element listy `list` używając tylko funkcji `Rest` i `First` i zaimplementowanej przez siebie wersji `Length`
- Proszę zaprogramować pełną funkcjonalność funkcji `Part` (nieobowiązkowe, wymaga użycia wzorca `BlankSequence`)

2 Zadanie 12

Hipoteza Collatza mówi, że ciąg zadany przez relację

$$a_{i+1} = f(a_i), \quad a_0 = n \quad (4)$$

$$f(n) = \begin{cases} n/2 & n \equiv 0 \pmod{2} \quad (n \text{ jest parzyste}) \\ 3n + 1 & n \equiv 1 \pmod{2} \quad (n \text{ jest nieparzyste}) \end{cases}. \quad (5)$$

zawsze zaczyna się na jedynce, t.j. dla dowolnego n istnieje takie $M(n)$, że $a_{M(n)} = 1$. Empiryczne sprawdzenie tej hipotezy polega na aplikowaniu (4) do momentu, gdy uzyskamy liczbę 1. Proszę sprawdzić, że dla $n = 27$ ciąg kończy się na jedynce (ile kroków do tego potrzeba?).

Przykład: zaczynając od $n = 11$ mamy $a_0 = 11, a_1 = 34, a_2 = 17, \dots, a_{14} = 1$.

(nieobowiązkowe) Proszę narysować graf skierowany obrazujący iteracje dla pierwszego tysiąca liczb ($n \in [2 : 1000]$).