Zadanie 1. Zapoznaj się z poniższym algorytmem NWD, a następnie zaproponuj schemat blokowy algorytmu oraz jego implementację (rozwiązanie należy zaproponować w formie iteracyjnej i rekurencyjnej).

Algorytm Euklidesa służy do wyznaczania największego wspólnego dzielnika dwóch liczb całkowitych. Największy wspólny dzielnik dwóch liczb a i b, to taka liczba, która dzieli te liczby bez reszty i jest ona możliwie największa. Można go zastosować do skracania ułamków lub wyznaczenia najmniejszej wspólnej wielokrotności NWW.

Wersja I – nieoptymalna postać algorytmu, polega na wybraniu większej z dwóch liczb i zamianie na różnicę większej i mniejszej. Czynność powtarzamy do momentu uzyskania dwóch takich samych wartości.

1. Podaj a, b
2. Jeżeli a=b to idź do K5
3. Jeżeli a>b, $a \leftarrow a - b$
4. else b ← b − a
5. Wypisz a
6. Koniec

Wykonaj analizę dla danych wejściowych NWD(12,18), NWD(28,24).

Wersja II - zoptymalizowany algorytm Euklidesa dla dwóch liczb naturalnych a i b. W każdym przejściu pętli wykonujemy dwie operacje: $a = b$ oraz $b = a \; mod \; b$. Czynności te powtarzamy do momentu, gdy zmienna b osiągnie wartość zero. Zmienna a będzie przechowywać wtedy największy wspólny dzielnik liczb podanych na wejściu.

Zaproponuj algorytm dla wersji iteracyjnej oraz rekurencyjnej.

Zadanie2. Dana jest następująca funkcja rekurencyjna

```
funkcja wynik(i)
        jeżeli i<3
                zwróć 1 i zakończ;
        w przeciwnym razie
                jeżeli i mod 2 =0;
                        zwróć wynik(i-3) + wynik(i-1)+1;
                w przeciwnym razie
                        zwróć wynik(i-1) mod 7
```

Uzupełnij poniższą tabelę:

i	wynik(i)	Ilość wywołań funkcji bez wywołania głównego
2	2	0
3		
4		
5		
6		
7		
8		

Zadanie3. Dana jest funkcja f określona wzorem rekurencyjnym:

$$\begin{cases} f(1) = 4 \\ f(n+1) = \dfrac{1}{1\text{-}f(n)}, \text{dla } n \geq 1 \end{cases}$$

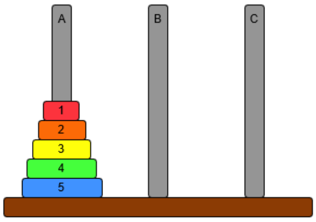
wtedy:

1	$f(8) = \frac{1}{3}$	P	F
2	$f(9) = \frac{3}{4}$	P	F
3	$f(10) = 4$	P	F
4	$f(100) = -\frac{1}{3}$	P	F

Zadanie 4. Zaproponuj rekurencyjny algorytm zamiany liczby dziesiętnej na binarną. Należy zaprojektować schemat blokowy oraz implementacje.

Zadanie 5. Zapoznaj się z problemem wieży Hanoi, a następnie zaproponuję listę kroków, schemat blokowy oraz implementację algorytmu.

Prosta zabawka dziecięca — na patyku nanizanych jest pewna liczba krążków tak, że na większym zawsze leży krążek mniejszy. Zadaniem naszym jest umieszczenie wszystkich krążków na sąsiednim „patyku" (korzystając z jednego tylko „patyka" pomocniczego) w tej samej kolejności. Podczas każdego ruchu pamiętać trzeba, że krążek większy nie może znaleźć się nigdy na krążku mniejszym.



Idea algorytm:

- Gdy krążek jest tylko jeden — problem nie istnieje (przenosimy go z patyka, na którym się znajduje na patyk docelowy).
- Dla dwu krążków problem jest banalny (najmniejszy krążek przenosimy na roboczy, większy na docelowy i ponownie najmniejszy na docelowy).

Przykład. Problem dla trzech krążków: można go podzielić na trzy zadania:

1. Przenosimy dwa „górne" krążki na „trzeci patyczek" (patyk roboczy).
2. Przeniesienie największego krążka na „patyczek drugi" (docelowy).
3. Ponowne przeniesienie dwu krążków z „roboczego" na krążek największy (znajdujący się na patyku docelowym)…

Ogólnie, będzie jakoś tak: A — patyk, na którym są wszystkie krążki, B — patyk docelowy, a C — patyk roboczy).

Procedura jest następująca:

- Przenieśmy z A na C N−1 krążków (używamy do tego procedury); B jest patykiem roboczym.
- Pozostały krążek (największy! — ale z czego to wynika?) przenieśmy z A na B (miejsce docelowe).
- Do pozostałych (N − 1) krążków, które znajdują się na patyku C, zastosujmy powyższy algorytm (patyk B wykorzystujemy jako roboczy, bo na samym spodzie znajduje się krążek największy). Zatem ruch wygląda tak: przenieś N − 1 krążków z C na B używając A jako patyka roboczego.
- powyższą procedurę należy powtarzać aż do zakończenia zadania.

Algorytm generuje jedynie podpowiedzi w formie $\alpha \rightarrow \beta$ oznaczające „weź krążek z patyka α i przenieś go na patyk β.