

Zestaw ćwiczeń do przedmiotu „**Algorytmy i złożoność**” dla studiów
dziennych i zaocznych

Rok akad. 2021/22

Ćwiczenia 3 – Algorytmy rekurencyjne

gr. | gr.

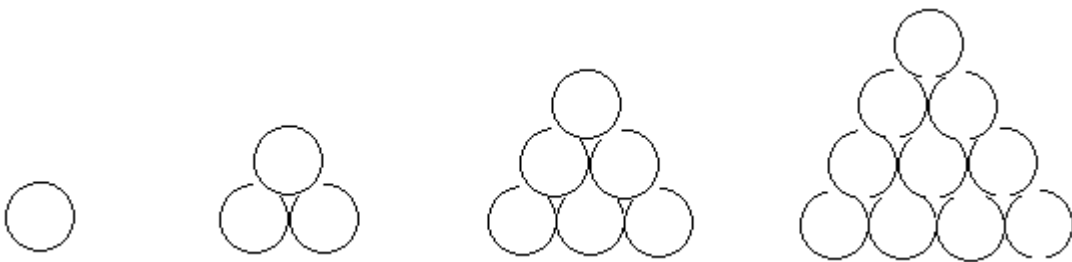
1. Zapisz algorytm rekurencyjny obliczający silnię podanej liczby **n**.
2. Zapisz algorytm rekurencyjny, obliczający największy wspólny dzielnik dla dwóch podanych liczb **a** i **b**.
3. Zapisz algorytm rekurencyjny obliczający n-ty wyraz poniższego ciągu:

$$a_n = \begin{cases} 1, & \text{dla } n = 1 \\ 0, 5, & \text{dla } n = 2 \\ -a_{n-1} \cdot a_{n-2}, & \text{dla } n > 2 \end{cases}$$

4. Zapisz algorytm rekurencyjny obliczający n-ty wyraz poniższego ciągu:

$$a_n = \begin{cases} -1, & \text{dla } n = 1 \\ -a_{n-1} \cdot n - 3 & \text{dla } n > 1 \end{cases}$$

5. Zapisz algorytm rekurencyjny konwertujący liczbę **a** (a jest argumentem funkcji) z systemu dziesiętnego na ósemkowy.
6. Dodatkowo zmodyfikuj powyższy algorytm tak, aby możliwe było zdefiniowanie poprzez dodatkowy argument systemu, na który wartość **a** ma zostać przekonwertowana.
7. Zapisz algorytm rekurencyjny obliczający sumę cyfr liczby naturalnej.
8. Z monet o tych samych nominałach tworzymy kolejne trójkąty „równoboczne”. Zapisz algorytm rekurencyjny, który obliczy wartość monet w n-tym trójkącie.



Metodyka:

Algorytmy zapisywać w postaci funkcji stosując podstawowe notacje języka **Java**. TO NIE JEST NAUKA PROGRAMOWANIA. W pierwszej części semestru stosujemy do zapisu algorytmu podstawowe typy proste i tablice o stałych rozmiarach, instrukcje warunkową, pętle **while** i **do while**, pętle **for**, rekurencje. Używanie komputera celem zapisu i testowania algorytmu nie jest praktykowane – powoduje stratę czasu i niepotrzebnie rozprasza studentów, którzy powinni skupić się na sposobie rozwiązywania problemu i własnościach poznawanych struktur danych. Poprawność algorytmów badać wyłącznie drogą, opartego na logicznym wnioskowaniu, rozumowania.

Zwrócić uwagę, na uzasadnienie wyboru rodzaju pętli iteracyjnej, a także określenie: co jest daną i wynikiem algorytmu, jakie są typy danych wejściowych i wyjściowych algorytmu i

jakie warunki (*asercja początkowa i końcowa*) spełniają te dane, a także - co jest warunkiem stopu pętli iteracyjnych, lub procesu rekurencyjnego i czy jest on spełniony?