

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

rok akad. 2022/23

**Ćwiczenia 3** – Algorytmy rekurencyjne

1. Zapisać iteracyjny, a następnie **rekurencyjny**, algorytm który bada czy we wczytywanym kolejno z konsoli, ciągu liczb całkowitych (bez użycia tablicy), zakończonym wczytaniem **0**, występuje przynajmniej jeden raz, zadana wartość **x**.

Dana jest tablica **unikalnych** liczb całkowitych, zadeklarowana **int t[N]**.

2. Zapisać algorytm w postaci funkcji iteracyjnej, badającej czy w tablicy występuje przynajmniej jeden raz zadana wartość **x**. Następnie algorytm ten przekształcić do postaci rekurencyjnej.
3. Zapisać iteracyjny (był), a następnie **rekurencyjny**, algorytm który zwraca indeks wystąpienia zadanej wartości **x** w tablicy, lub wartość **-1** gdy element o takiej wartości w tablicy nie występuje.
4. Zapisać iteracyjny, a następnie **rekurencyjny**, algorytm który liczy sumę elementów tablicy.

Liczbę całkowitą (np. 308) zapisano w tablicy liczb całkowitych, przechowującej kolejne cyfry liczby.

5. Zapisać **rekurencyjny** algorytm obliczający wartość numeryczną liczby wpisanej do tablicy.

Dana jest **tablica znakowa** o dostatecznie dużych rozmiarach, która przechowuje **napis** (ciąg znaków Unicode), zakończony znakem o kodzie 0.

6. Zapisz **rekurencyjny** algorytm, który wypisuje na konsolę **odwrotnie**, przechowywany w tablicy znakowej napis. ALGORYTM BEZOGONOWY. Skorzystać z metod **length()**, oraz **charAt(index)** klasy *String*.
7. Zapisz **rekurencyjny** algorytm badający czy w napisie występuje przynajmniej jeden raz zadany znak **c**.
8. Zapisz rekurencyjny algorytm, który zlicza, ile razy występuje w tablicy zadany znak **c**.
9. Dany jest poniższy **rekurencyjny** algorytm operujący na tablicy liczb całkowitych **int t[N]**.

```
int fun( int i, int j)
{ if(i<N) return fun(i+1, j+1);
  else return j;
}
```

Odpowiedz, jaką wartość otrzyma zmienna **x** w wyniku wywołania tego algorytmu w następujący sposób **int x = fun( 0, 0 );** Odpowiedź szczegółowo uzasadnij.

10. Zapisać **rekurencyjny** algorytm, który wypisuje w kolejnych wierszach na konsolę cyfry zadanej liczby całkowitej **x**. Następnie algorytm ten przekształcić do postaci iteracyjnej. ALGORYTM BEZOGONOWY.

11. Zapisać **rekurencyjny** algorytm, badający czy wpisany do tablicy tekst jest **palindromem**. Tekst wypełnia całą tablicę, tzn. długość tekstu jest równa ilości elementów tablicy. Pomysł na rozwiązanie zadania – ustawić indeks **i** na pierwszym elemencie tablicy, oraz indeks **j** na ostatnim elemencie tablicy. Następnie indeksy przesuwają w każdym kroku: **i** w prawo, a **j** w lewo. Omówić warunki zatrzymania procesu rekurencji.

Kilka przykładowych palindromów:

Zakopane na pokaz.

Kobyła ma mały bok.

Może jutro ta dama da tortu jeżom.

Kat Pąg Izidor świni w środy z Iką Ptak.