**Wytyczne:**

1. Ćwiczenia można wykonać w grupach maksymalnie 4-osobywych.
2. Algorytmy mogą być napisane w dowolnym języku programowania, preferowany jest jednak C#
3. Warunkiem zaliczenie jest wyjaśnienie prowadzącemu każdego z algorytmów
4. **Ćwiczenie nr 1 – algorytm wyszukiwania liniowego**
5. Zaimplementować algorytm wyszukiwania liniowego.
6. Wczytać z zewnętrznego pliku dane do wyszukiwania zawierające min. 1000 słów (np. artykuł internetowy ) w języku innym niż polski. Wskazać po ilu krokach wyszukają się dowolne słowo na literę „a”, „c”, „d”, „m”, „w”, „z”.
7. Wczytać z zewnętrznego pliku dane do wyszukiwania zawierające min. 1000 słów (np. artykuł internetowy ) w języku innym niż polski i innym niż w pkt. 1b). Wskazać po ilu krokach wyszuka się ostatnie słowo z wczytanego tekstu.
8. **Ćwiczenie nr 2 – algorytm wyszukiwania binarnego**
9. Zaimplementować algorytm wyszukiwania binarnego.
10. Wczytać z zewnętrznego pliku dane do wyszukiwania zawierające min. 1000 słów (np. artykuł internetowy ) w języku innym niż polski. Wskazać po ilu krokach wyszukają się dowolne słowo na literę „a”, „c”, „d”, „m”, „w”, „z”.
11. Wczytać z zewnętrznego pliku dane do wyszukiwania zawierające min. 100 słów (np. artykuł internetowy ) w języku innym niż polski i innym niż w pkt. 2b). Wskazać po ilu krokach wyszuka się ostatnie słowo z wczytanego tekstu.
12. **Ćwiczenie nr 3 – – algorytm sortowania przez wybór**
13. Zaimplementować algorytm sortowania przez wybór
14. Wczytać z zewnętrznego pliku dane do wyszukiwania zawierające min. 1000 słów (np. artykuł internetowy ) w języku innym niż polski. Wskazać po ilu krokach słowa zostaną posortowane.
15. Wczytać z zewnętrznego pliku dane do wyszukiwania zawierające min. 1000 słów (np. artykuł internetowy ) w języku innym niż polski i innym niż w pkt. 3b). Wskazać po ilu krokach słowa zostaną posortowane.
16. **Ćwiczenie nr 4 – algorytm sortowania szybkiego**
17. Zaimplementować algorytm sortowania szybkiego
18. Wczytać z zewnętrznego pliku dane do wyszukiwania zawierające min. 1000 słów (np. artykuł internetowy ) w języku innym niż polski. Wskazać po ilu krokach słowa zostaną posortowane.
19. Wczytać z zewnętrznego pliku dane do wyszukiwania zawierające min. 1000 słów (np. artykuł internetowy ) w języku innym niż polski i innym niż w pkt. 3b). Wskazać po ilu krokach słowa zostaną posortowane.
20. **Ćwiczenie nr 5 – algorytmy operacji na macierzy**
21. Zaimplementować algorytm dodawania dwóch macierzy o wymiarach . Sprawdzić ile kroków jest wykonywanych w przypadku dodawania macierzy o rozmiarach:
22. Zaimplementować algorytm mnożenia dwóch macierzy . Sprawdzić ile kroków jest wykonywanych w przypadku mnożenia macierzy o rozmiarach:
23. Zaimplementować algorytm transpozycji macierzy . Zaprezentować działanie.
24. **Ćwiczenie nr 6 – ciąg Fibbonaciego z różnymi podejściami**
25. Dla liczby wynoszącej , zaokrąglonej do liczby całkowitej, np. =357, zaimplementować:
    1. algorytm rekurencyjny znajdowania ciągu Fibbonaciego
    2. algorytm wykorzystujący tzw. memoizację
    3. algorytm wykorzystujący podejście Bottom-up
    4. dla każdego z tych przypadków wskazać liczbę kroków wykonania się algorytmu
26. **Ćwiczenie nr 7 – przeszukiwanie grafu**
27. narysować graf nieskierowany składających się co najmniej z 10 wierzchołów
28. zaimplementować algorytm BFS przechodzenia przez graf
29. zaimplementować algorytm DFS przechodzenia przez graf.
30. Wskazać liczbę wykonanych w pkt. b) i c) kroków
31. **Ćwiczenie nr 8 – wyszukiwanie cykli w grafie**
32. narysować graf nieskierowany składających się co najmniej z 10 wierzchołów i zawierający cykle
33. zaimplementować algorytm odnajdywania cyklu zgodnie z metodą BFS
34. zaimplementować algorytm odnajdywania cyklu zgodnie z metodą DFS
35. Wskazać liczbę wykonanych w pkt. b) i c) kroków
36. **Ćwiczenie nr 9 – operacje na kolejkach**
37. Wczytać dowolny i posortowany tekst z zewnętrznego pliku zawierającego co najmniej 1000 słów do kolejki
38. usunąć z kolejki elementy od 100 do 200
39. wyświetlić zawartość kolejki
40. **Ćwiczenie nr 10 – algorytmy geometryczne**
41. Zaimplementować algorytm SoundEx
42. Potrafić wyjaśnić jak działa i do czego służy
43. Za jego pomocą policzyć podobieństwa oddzielnie dla imion oraz nazwisk w swojej grupie.
44. Zaimplementować algorytm wyznaczający długość Levenshteina oraz za jego pomocą porównać oddzielnie imiona i nazwiska w swojej grupie.
45. **Ćwiczenie nr 11 – algorytmy geometryczne**
46. Zaimplementować algorytmy:
    1. Wyszukiwania punktu środowego wektora
    2. Obliczania kąta nachylenia wektora
    3. Znajdywania minimalnej liczby linii przechodzących przez zadane, wszystkie N punktów w przestrzeni 2D oraz zadane punkty
    4. Obliczania powierzchni zadanego trójkąta
    5. Sprawdzający, czy linia przechodzi, nie przechodzi lub styka się z okręgiem

Obraz zawierający szkic, design

Zawartość wygenerowana przez sztuczną inteligencję może być niepoprawna.