



Sortowanie pozycyjne słów

- [Wprowadzenie](#)
- [Przeczytaj](#)
- [Schemat interaktywny](#)
- [Sprawdź się](#)
- [Dla nauczyciela](#)



Na co dzień często mamy do czynienia ze zbiorami posortowanych słów – najczęściej w kolejności alfabetycznej. W taki sposób np. posortowane są wyrazy w słowniku i nazwiska uczniów w dzienniku szkolnym. W tym e-materiale dowiesz się, w jaki sposób uporządkować zbiór słów, wykorzystując koncepcję algorytmu sortowania pozycyjnego.

Jak już wiesz, sortowanie pozycyjne jest specjalnym algorytmem sortowania, który pozwala na szybkie porządkowanie danych spełniających konkretne warunki – jest to jeden z niewielu algorytmów sortowania o złożoności liniowej. Przeczytasz o nim również w e-materiałach:

- [Sortowanie pozycyjne dat](#),
- [Sortowanie pozycyjne liczb](#).

Implementacje sortowania pozycyjnego słów przedstawiamy w e-materiałach:

- [Sortowanie pozycyjne słów w języku C++](#),
- [Sortowanie pozycyjne słów w języku Java](#),
- [Sortowanie pozycyjne słów w języku Python](#).

Więcej zadań? Sięgnij do: [Sortowanie pozycyjne słów – zadania maturalne](#).

Twoje cele

- Posortujesz zbiór słów za pomocą algorytmu sortowania pozycyjnego.
- Przeanalizujesz przypadek optymistyczny i pesymistyczny algorytmu sortowania pozycyjnego słów.
- Wykonasz ćwiczenia utrwalające umiejętność sortowania pozycyjnego słów.

Przeczytaj

Sortowanie pozycyjne słów

W wypadku sortowania pozycyjnego słów, podobnie jak przy sortowaniu pozycyjnym dat i liczb, nie porównuje się ze sobą wartości wprost, tylko ich kolejne elementy składowe, zaczynając do najmniej znaczącego.

Przeanalizujmy przykład sortowania zbioru słów za pomocą algorytmu sortowania pozycyjnego. Naszym zadaniem jest uporządkowanie (od A do Z) następującego zbioru słów:

- DEC,
- DAC,
- CAD.

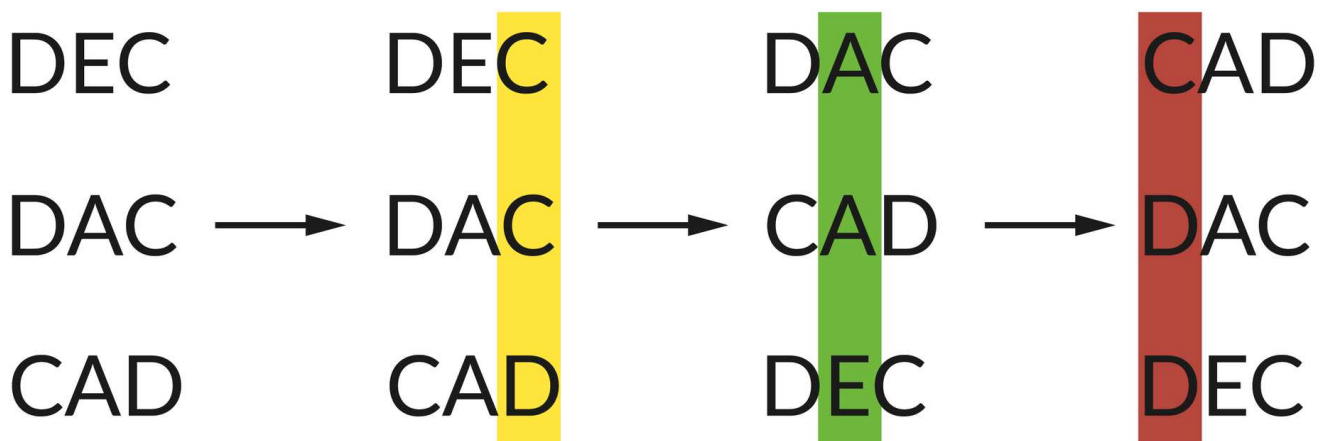
Zacznijmy od przypisania określonej liczby naturalnej każdej literze alfabetu łacińskiego. Liczby te będą pełniły rolę „odpowiedników”, za pomocą których określimy, czy dane dwie litery są na odpowiednich miejscach.

W przedstawionej tabeli kolejnym literom alfabetu łacińskiego przypisano kolejne liczby naturalne.

Litera	Przypisana liczba
A	1
B	2
C	3
D	4
E	5
F	6
G	7
H	8
I	9
J	10
K	11
L	12
M	13

Litera	Przypisana liczba
N	14
O	15
P	16
Q	17
R	18
S	19
T	20
U	21
V	22
W	23
X	24
Y	25
Z	26

Następnie należy posortować słowa według kolejnych pozycji (wykorzystamy [porządek leksykograficzny](#)), zaczynając od pozycji najmniej znaczącej, za pomocą [stabilnego](#) algorytmu sortowania, np. sortowania bąbelkowego, kubełkowego, przez wstawianie. Grafika przedstawia, jak wygląda zbiór po sortowaniach na kolejnych pozycjach. Kolorami oznaczono, według której pozycji odbywało się sortowanie.



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Przeanalizujmy powyższy przykład bardziej szczegółowo, a jako stabilny algorytm sortowania wybierzmy algorytm sortowania bąbelkowego.

Ważne!

Algorytm sortowania bąbelkowego nie jest jedynym stabilnym algorytmem sortowania. Do tej grupy algorytmów zaliczamy również m.in. sortowanie przez zliczanie, sortowanie

przez wstawianie, sortowanie kubełkowe, sortowanie przez scalanie.

Numery kolejnych pozycji (zaczynając od najmniej znaczącej) oznaczono kolejno 0, 1, 2:

0 1 2
D E C

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Dla **pozycji 2**:

0 1 2
D E C
D A C
C A D

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

1. Czy $3 > 3$? ($C > C$) – porównywane są wartości liczbowe przypisane do odpowiednich liter, ponieważ algorytm sortowania jest stabilny, nie następuje zamiana.
2. Czy $3 > 4$? ($C > D$) – porównywane są kolejne liczby, jednak zamiana nie następuje, ponieważ chcemy posortować zbiór niemalejąco.
3. Po przejściu wszystkich liter na pozycji 2 nie nastąpiła zmiana, w związku z czym można zakończyć wykonywanie algorytmu sortowania bąbelkowego i przejść do pozycji nr 1.

Dla **pozycji 1**:

0	1	2
D	A	C
C	A	D
D	E	C

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

1. Czy $5 > 1$? ($E > A$) – tak, wykonywana jest więc zamiana słów.
2. Czy $5 > 1$? ($E > A$) – tak, wykonywana jest więc zamiana słów.
3. Po kolejnym przejściu wszystkich liter na pozycji 1 nie nastąpiła zmiana, w związku z czym można zakończyć wykonywanie algorytmu sortowania bąbelkowego i przejść do pozycji nr 0.

Dla **pozycji 0**:

0	1	2
C	A	D
D	A	C
D	E	C

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

1. Czy $4 > 3$? ($D > C$) – tak, następuje zamiana słów.
2. Czy $4 > 4$? ($D > D$) – ponieważ algorytm sortowania jest stabilny, nie następuje zamiana.
3. Po przejściu wszystkich liter na pozycji 0 nie nastąpiła zmiana, w związku z czym można zakończyć wykonywanie algorytmu sortowania bąbelkowego i zakończyć algorytm.

Pseudokod

Podczas sortowania pozycyjnego danego zbioru słów postępujemy zgodnie z następującym algorytmem.

Specyfikacja:

Dane:

- Z – tablica słów długości n
- n – liczba naturalna; długość słów zawartych w tablicy Z

Wynik:

Funkcja zwraca posortowaną tablicę Z .

```
1 funkcja sortowaniePozycyjne(Z, n)
2   dla  $i = n-1, n-2, \dots, 0$  wykonuj
3     posortuj stabilnie tablicę  $Z$  według znaków na pozycji  $i$ 
```

W przedstawionym pseudokodzie zostało przyjęte, że każdy element ze zbioru Z składa się z n znaków. Pozycja 0 jest najbardziej znaczącą pozycją, natomiast pozycja $n - 1$ – najmniej znacząca.

Złożoność obliczeniowa

Sortowanie według kolejnych pozycji (liter) przeprowadzane jest w czasie liniowym $O(n)$. Sortowanie wykonywane jest tyle razy, z ilu liter składają się słowa z danego zbioru.

Słownik

algorytm stabilny

algorytm, który dla dwóch równych elementów w zbiorze danych wejściowych zachowuje ich kolejność

porządek leksykograficzny

sposób porządkowania ciągów w sposób analogiczny do porządku alfabetycznego

Schemat interaktywny

Polecenie 1

Zapisz algorytm sortowania pozycyjnego słów w kolejności rosnącej, wykorzystujący algorytm sortowania bąbelkowego dla danych pozycji. Jeśli chcesz powtórzyć wiadomości ze szkoły podstawowej, zrób to za pomocą schematu blokowego. W przeciwnym razie wykorzystaj wybrany język programowania.

Przetestuj działanie programu dla następującego zbioru słów:

```
1 {"BABA", "SOFA", "ABBA", "COFA", "CYKA", "BOBO"}
```

Specyfikacja:

Dane:

- *lista* – zbiór n -literowych słów
- n – liczba naturalna; długość słów ze zbioru *lista*

Wynik:

Na standardowym wyjściu wyświetlane są wszystkie elementy tablicy *lista* posortowane rosnąco.

Zapisz algorytm, wykorzystując wybrany język programowania.

```
1
```

Polecenie 2

Dodaj do swojego programu komentarze tak, żeby był zrozumiały dla osoby, która nie potrafi programować.

Sprawdź się

Pokaż ćwiczenia:   

Ćwiczenie 1



Zaznacz, które z poniższych algorytmów sortowania są **stabilnymi** algorytmami sortowania.

algorytm sortowania szybkiego

algorytm sortowania bąbelkowego

algorytm sortowania przez zliczanie

algorytm sortowania kubełkowego

Ćwiczenie 2



Wstaw w zdanie odpowiednie słowa, aby stanowiło poprawną definicję stabilnego algorytmu sortowania.

Stabilny algorytm sortowania to taki, który o wartości, pozostawi w kolejności.

różnej

niezmienionej

takiej

podobnej

samej

elementy

Ćwiczenie 3



Wskaż, dla którego zbioru Z algorytm sortowania pozycyjnego słów sprawdzi się najlepiej.

$Z = \{EF, EL, KA, TYLKO, WOJSKO\}$

$Z = \{TERAZ, JUTRO, POJUTRZE, PRZEDWCZORAJ\}$

$Z = \{AB, LOKI, MITOLOGIA, POLSKA\}$

$Z = \{ŻABA, FOKA, ABBA, BAKU, JAVA\}$

Ćwiczenie 4



Zapisz, jak będzie wyglądał zbiór $Z = [ABBA, KARS, JAKI, POLE]$ po pierwszej iteracji sortowania pozycyjnego w kolejności nierosnącej.

Ćwiczenie 5



Dokończ zdanie.

Algorytm sortowania pozycyjnego najlepiej sprawdza się dla zbioru słów...

- o nieparzystej liczbie elementów.
- w którym słowa mają tę samą liczbę liter.
- o parzystej liczbie elementów.

Ćwiczenie 6



Zapisz, jak będzie wyglądał zbiór $Z = [GRYFY, POLEK, NOMAD, PROGI]$ po pierwszej iteracji sortowania pozycyjnego w kolejności niemalejącej.

Ćwiczenie 7



Zaznacz, czy zdanie jest prawdziwe.

Idea algorytmu sortowania pozycyjnego jest taka sama w przypadku sortowania liczb, dat czy słów.

- fałsz
- prawda

Ćwiczenie 8



Wskaż, co, zgodnie z algorytmem sortowania pozycyjnego, powinno się stać w przypadku, gdy algorytm porównuje dwa elementy o tej samej wartości.

- Elementy powinny zostać umieszczone odpowiednio na początku i końcu zbioru.
- Elementy zbioru powinny zostać zamienione.
- Elementy zbioru powinny pozostać bez zmian.

Dla nauczyciela

Autor: Maurycy Gast

Przedmiot: Informatyka

Temat: Sortowanie pozycyjne słów

Grupa docelowa:

Szkoła ponadpodstawowa, liceum ogólnokształcące, technikum, zakres rozszerzony

Podstawa programowa:

Cele kształcenia – wymagania ogólne

I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

Treści nauczania – wymagania szczegółowe

I. Rozumienie, analizowanie i rozwiązywanie problemów.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

3) objaśnia dobrany algorytm, uzasadnia poprawność rozwiązania na wybranych przykładach danych i ocenia jego efektywność;

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

3) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu, uruchamianiu i testowaniu programów;

I + II. Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

3) objaśnia, a także porównuje podstawowe metody i techniki algorytmiczne oraz struktury danych, wykorzystując przy tym przykłady problemów i algorytmów, w szczególności:

i) struktury dynamiczne: stos, kolejka, lista (do realizacji algorytmu: ONP, symulacji problemu Flawiusza, sortowania leksykograficznego),

Kształtowane kompetencje kluczowe:

- kompetencje cyfrowe;
- kompetencje osobiste, społeczne i w zakresie umiejętności uczenia się;
- kompetencje matematyczne oraz kompetencje w zakresie nauk przyrodniczych, technologii i inżynierii.

Cele operacyjne (językiem ucznia):

- Posortujesz zbiór słów za pomocą algorytmu sortowania pozycyjnego.
- Przeanalizujesz przypadek optymistyczny i pesymistyczny algorytmu sortowania pozycyjnego słów.
- Wykonasz ćwiczenia utrwalające umiejętność sortowania pozycyjnego słów.

Strategie nauczania:

- konstruktywizm;
- konektywizm.

Metody i techniki nauczania:

- dyskusja;
- rozmowa nauczająca z wykorzystaniem multimediu i ćwiczeń interaktywnych;
- metody aktywizujące.

Formy pracy:

- praca indywidualna;
- praca w parach;
- praca w grupach;
- praca całego zespołu klasowego.

Środki dydaktyczne:

- komputery z głośnikami, słuchawkami i dostępem do internetu;
- zasoby multimedialne zawarte w e-materiale;
- tablica interaktywna/tablica, pisak/kreda.

Przebieg lekcji

Przed lekcją:

1. **Przygotowanie do zajęć.** Nauczyciel loguje się na platformie i udostępnia e-materiał: „Sortowanie pozycyjne słów”. Nauczyciel prosi uczniów o zapoznanie się z treściami w sekcji „Przeczytaj”.

Faza wstępna:

1. Nauczyciel wyświetla uczniom temat zajęć oraz cele. Prosi, by na ich podstawie uczniowie sformułowali kryteria sukcesu.
2. **Rozpoznanie wiedzy uczniów.** Uczniowie tworzą pytania dotyczące tematu zajęć, na które odpowiedzą w trakcie lekcji.

Faza realizacyjna:

1. **Praca z tekstem.** Nauczyciel ocenia, na podstawie informacji na platformie, stan przygotowania uczniów do zajęć. Jeżeli jest ono niewystarczające, prosi o ciche zapoznanie się z treścią w sekcji „Przeczytaj”.
2. **Praca z multimediami.** Nauczyciel wyświetla zawartość sekcji „Schemat interaktywny”, czyta treść polecenia nr 1 i omawia kolejne kroki prowadzące do rozwiązania.
3. **Ćwiczenie umiejętności.** Liga zadaniowa - uczniowie wykonują indywidualnie na czas ćwiczenia nr 1-5 z sekcji „Sprawdź się”, a następnie omawiają zadania na forum.

Faza podsumowująca:

1. Na koniec zajęć nauczyciel raz jeszcze wyświetla na tablicy temat lekcji i cele zawarte w sekcji „Wprowadzenie”. W odniesieniu do ich realizacji dokonuje szczegółowej oceny rozwiązania zastosowanego przez wybraną osobę.
2. Na koniec zajęć nauczyciel prosi uczniów o rozwinięcie zdania: „Na dzisiejszych zajęciach nauczyłam/łem się jak...”.

Praca domowa:

1. Uczniowie wykonują ćwiczenia 6-8 z sekcji „Sprawdź się”.

Wskazówki metodyczne:

- Treści w sekcji „Przeczytaj” można wykorzystać jako podsumowanie i utrwalenie wiedzy uczniów.