

Znajdowanie lidera w zbiorze

Liderem będziemy nazywać element zbioru występujący w nim więcej niż $\lceil n/2 \rceil$ razy, gdzie n oznacza ilość elementów zbiorze. Innymi słowy liderem jest element, którego liczba wystąpień jest większa, od sumy liczby wystąpień pozostałych elementów.

W zbiorze { **9** 2 **9** 5 **9** 7 **9** } liderem jest 9, ponieważ 9 występuje w zbiorze 4 razy, a wszystkie pozostałe elementy występują w sumie 3 razy.

Narzucającym się rozwiązaniem tego problemu jest znalezienie lidera przy pomocy wcześniej opisanych algorytmów wyszukiwania najczęściej pojawiającego się elementu. Można to wykonać w sposób następujący:

Wyszukujemy najczęściej występujący element. Sprawdzamy, czy ilość jego wystąpień jest większa od $\lceil n/2 \rceil$. Jeśli tak, znaleźliśmy lidera. W przeciwnym razie lider w zbiorze nie występuje.

Algorytm posiada klasę czasowej złożoności obliczeniowej $O(n^2)$ lub $O(n + m)$ w zależności od wyboru algorytmu wyznaczającego najczęstszy element zbioru.

Program wypełnia 20 elementowy zbiór pseudolosowymi liczbami z zakresu od 0 do 9, a następnie poszukuje w nim lidera.

```
# -*- coding: cp1250 -*-
```

```
# Wyszukiwanie lidera
```

```
import random
```

```
N = 20      # liczba elementów w zbiorze
```

```
WMAX = 9    # górny zakres elementów
```

```
d = []
```

```
# Generujemy zbiór liczb pseudolosowych
```

```
wn = random.randint(0, WMAX)
```

```
for i in range(N): d.append(wn)
```

```
for i in range(4 + N / 2):
```

```
    d[random.randint(0, N - 1)] = random.randint(0, WMAX)
```

```
# Wyszukujemy najczęstszy element
```

```
wn, ln, i = d[0], 1, 0
while i < N - ln:
    licznik = 1
    for j in range(i + 1, N):
        if d[i] == d[j]: licznik += 1
    if licznik > ln: wn, ln = d[i], licznik
    i += 1
wt = (ln > N / 2)
```

```
# Prezentujemy wyniki
```

```
for i in d:
    if wt and (wn == i):
        print "<%d>" % i,
    else:
        print " %d " % i,
print
print
if wt:
    print "lider = %d, ln = %d" % (wn, ln)
else:
    print "W zbiorze nie ma lidera"
print
print
```

```
# Gotowe
```

```
raw_input("Koniec, naciśnij dowolny klawisz...")
```