

ASD - klasówka 1
28.11.2001

1 – 8 pkt Rozważmy następujący algorytm sortowania:

```
krok := 0;
for i := 1 to n-1 do
  for j := i+1 to n do
    begin
      krok := krok + 1;
      if a[j-1] > a[j] then
        a[j-1] :=: a[j] {zamiana}
    end
```

Niech początkową wartością $a[i]$ będzie x_i .

a) Zaprojektuj efektywny algorytm, który dla każdego x_i oblicza krok, w którym x_i jest zamieniany pierwszy raz (0, gdy nie dzieje się to w żadnym kroku);

b) Oblicz, ile średnio **zamian** jest wykonywanych w powyższym algorytmie w modelu losowej permutacji.

2 – 6 pkt Wiadomo, że każdą nieujemną liczbę całkowitą można przedstawić jako sumę różnych liczb Fibonacciego, z których żadne dwie nie są kolejne. Taką reprezentację nazwiemy znormalizowaną reprezentacją Fibonacciego. Niech $L[0..k]$ będzie licznikiem, w którym liczby są zapisywane w znormalizowanej reprezentacji Fibonacciego (liczba zapisana w liczniku jest równa $\sum_{i=0}^k L[i] \cdot F_i$, gdzie k jest pozycją najbardziej znaczącej jedynek). Na początku $L[i] = 0$, dla każdego i , a $k = 0$. Opisz operację dodawania 1 do licznika L i zanalizuj koszt zamortyzowany wykonania takiej operacji.

3 – 3 pkt Pokaż, że każde drzewo wyszukiwań binarnych można przekształcić na AVL-drzewo za pomocą pojedynczych rotacji.

4 – 3 pkt Udowodnij, że każdy algorytm scalający uporządkowany ciąg n -elementowy z uporządkowaną parą elementów wymaga w pesymistycznym przypadku co najmniej $2 \log(n + 1)$ porównań.