

Portofoliu Algoritmi și complexitate

Pavel Andrei

Laborator 3

TODO 10. Considerăm o secvență $x = (x_0, \dots, x_{n-1})$ de n numere întregi, cu măcar un element pozitiv. O subsecvență a șirului este de forma $(x_i, x_{i+1}, \dots, x_j)$, cu $0 \leq i \leq j \leq n - 1$, iar suma subsecvenței este suma elementelor componentelor sale. Descrieți un algoritm pentru a determina subsecvența de sumă maximă. Estimați timpul de execuție al algoritmului, precizând operația dominantă.

```
1 print("Please insert the sequence: ", end="")
2 strs = input().split(' ')
3 v = [int(num) for num in strs]
4 n = len(v)
5 # python way of defining a n-dimensional list initialized to 0
6 sub_sums = [0 for i in range(0, n)]
7
8 best = (0, 0)
9 best_sum = 0
10 for i in range(0, n):
11     sub_sums[i] = v[i]
12     best_end_index = i
13     # after this loop v[j] = (sum from k=i to j of v[k])
14     for j in range(i+1, n):
15         sub_sums[j] = sub_sums[j-1] + v[j]
16         if sub_sums[j] > sub_sums[best_end_index]:
17             best_end_index = j
18     if sub_sums[best_end_index] > best_sum:
19         best_sum = sub_sums[best_end_index]
20         best = (i, best_end_index)
21
22 print("Best with a sum of", best_sum, "is: (x%d,...,x%d)" % best)
```

```
$ python3 src.py
Please insert the sequence: 1 2 3 4
Best with a sum of 10 is: (x0,...,x3)
$ python3 src.py
Please insert the sequence: 1 -2 3 4
Best with a sum of 7 is: (x2,...,x3)
$ python3 src.py
Please insert the sequence: 1 2 -3 4
Best with a sum of 4 is: (x0,...,x3)
```

Considerăm operația de baza ca fiind compararea elementelor tabloului v (liniile 16 și 18).

Notăm $T_l(n) :=$ timpul total de execuție al liniei l ; $T(n) :=$ timpul de execuție total.

$$T_{16}(n) = \sum_{i=0}^{n-1} \sum_{j=i+1}^{n-1} 1 = \sum_{i=0}^{n-1} ((n-1) - i) = n(n-1) - \sum_{i=0}^{n-1} i = n(n-1) - \frac{n(n-1)}{2} = \frac{n(n-1)}{2}$$

$$T_{18}(n) = \sum_{i=0}^{n-1} 1 = n$$

$$T(n) = \frac{n(n-1)}{2} + n = \frac{n(n+1)}{2}$$