

# Programarea Algoritmilor

## – SEMINAR NR. 3 –

### (grupele 132-134, 141-144)

1. Se citește un număr natural  $n$  și o listă  $L$  de numere întregi.
  - a) Să se găsească un prefix (primele  $x$  elemente) de sumă maximă pentru  $L$ .
  - b) Să se găsească un slice de lungime 3 și sumă maximă.
  - c) Să se găsească un slice de sumă maximă.

```
n = int(input("n = "))
L = [0]*n
for k in range(n):
    L[k] = int(input("elem: "))

# (a) --> S lista auxiliara, S[i] = L[0]+L[1]+...+L[i]
# S[0] = L[0]
# S[i] = S[i-1] + L[i]
S = [0]*n
S[0] = L[0]
for i in range(1, n):
    S[i] = S[i-1] + L[i]
print("\nL =", L, "\nS =", S)

suma_max = max(S)
poz = S.index(suma_max)
print("Suma maxima:", suma_max)
print("Prefixul de suma maxima:", *L[0:poz+1])

# (b) --> S lista auxiliara de lungime n-2, S[i]=L[i]+L[i+1]+L[i+2]
# S[0] = L[0] + L[1] + L[2]
# S[i] = S[i-1] - L[i-1] + L[i+2]
S = [0]*(n-2)
S[0] = sum(L[0:3])
for i in range(1, n-2):
    S[i] = S[i-1] - L[i-1] + L[i+2]
print("\nL =", L, "\nS =", S)

suma_max = max(S)
poz = S.index(suma_max)
print("Suma maxima:", suma_max)
print("Slice-ul de lungime 3 si suma maxima:", *L[poz:poz+3])

# (c) - programare dinamica --> S lista auxiliara de lungime n
# S[i] = suma maxima a unei subsecvente care se termina pe pozitia i
# S[0] = L[0]
# S[i] = L[i] + max(S[i-1], 0)
S = [0]*n
S[0] = L[0]
for i in range(1, n):
    S[i] = L[i] + max(S[i-1], 0)
print("\nL =", L, "\nS =", S)

suma_max = max(S)
poz_final = S.index(suma_max)
i = poz_final
while S[i] > L[i]: #cautam de unde incepe slice-ul pt suma_max
    i -= 1
print("Suma maxima:", suma_max)
print("Slice-ul de suma maxima:", *L[i:poz_final+1])
```