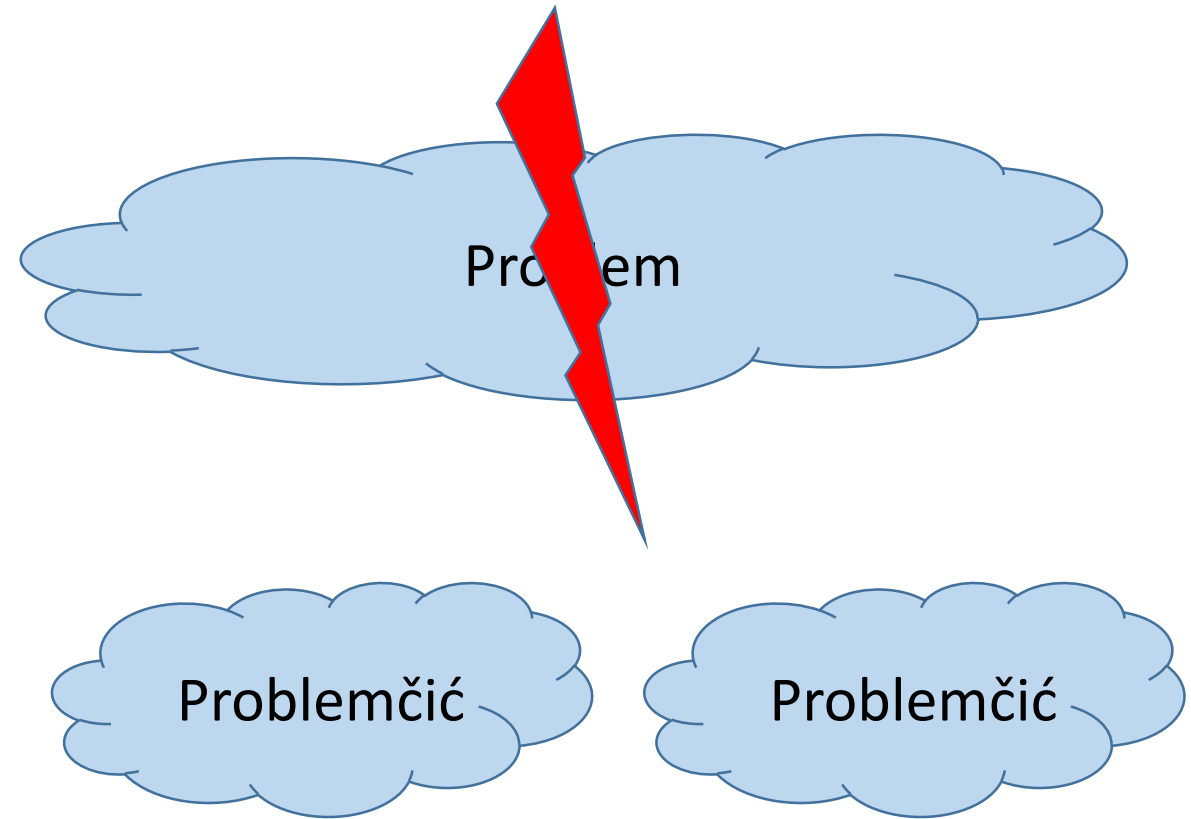


Algoritmi

L04. Analiza rekurzivnih algoritama. Rekurencije

"Zavadi pa vladaj"



Rekurzivno rešenje problema

Rešenje koje je definisano kao funkcija istog problema sa drugačijim parametrom!

Školski primer: **Fibonacci niz**

$$F(0) = 0$$

$$F(1) = 1$$

$$F(n) = F(n - 1) + F(n - 2)$$

Faktorijel

Nalaženje maksimuma

Odnos **iterativno – rekurzivno** rešenje

Dokaz tačnosti rekurzivnih rešenja

Matematička indukcija!

Primer: Pokazati da je $1 + 3 + 5 + \dots + (2n - 1) = n^2$

- Osnovni slučaj (**base case**) $n = 1$
- Rekurzivni slučaj (**recursive case**) iz tačnosti za n dokazujemo tačnost za $n+1$

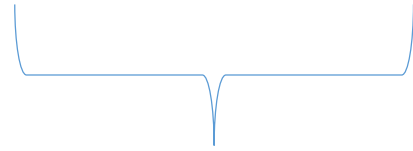
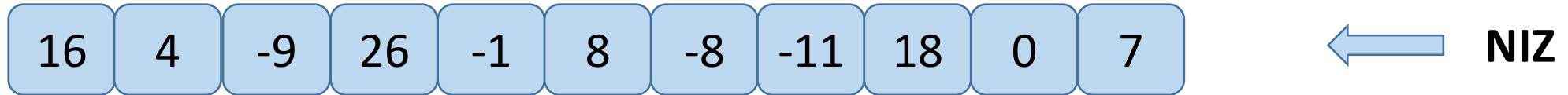
Opis vremena izvršenja rekurzivnih rešenja

Rekurencija – (ne)jednačina koja opisuje funkciju pomoću vrednosti iste funkcije na manjim vrednostima ulaza.

Primer:
$$T(n) = \begin{cases} \theta(1), & n = 1 \\ 2T\left(\frac{n}{2}\right) + \theta(n), & n > 1 \end{cases} \quad (\text{za Merge Sort})$$

Cilj: pronaći asimptotsko ponašanje vremena izvršenja $T(n) = \theta(f(n))$

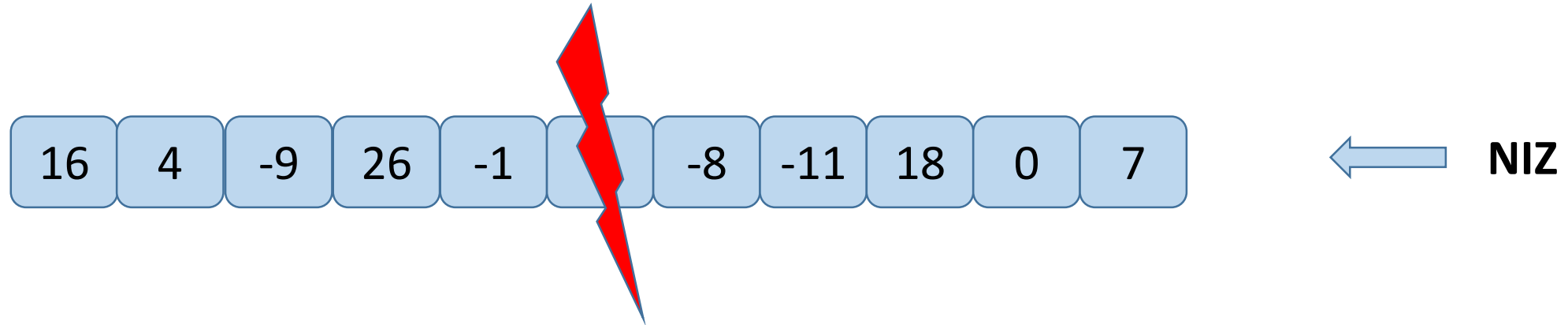
Još jedno rekurzivno rešenje problema





Problem: pronaći povezan podniz sa **najvećom sumom** elemenata.

→ Problem najvećeg podniza (maximum-subarray problem)

Još jedno rekurzivno rešenje problema



Rešenje je jedan od tri slučaja:

1. Ceo podniz je u levoj polovini niza,  **REKURZIVNO**
2. Ceo podniz je u desnoj polovini niza,  **REKURZIVNO**
3. Podniz sadrži srednji element i delom je u levoj, a delom u desnoj polovini.

$$T(n) = \begin{cases} \theta(1), & n = 1 \\ 2T\left(\frac{n}{2}\right) + \theta(n), & n > 1 \end{cases}$$

Kako rešiti rekurentnu jednačinu?

1. Metoda zamene
2. Metoda stabla rekurzije
3. Master metoda

Metoda zamene

1. Pretpostavimo formu rešenja.
2. Matematičkom indukcijom pronađemo konstante i dokažemo tačnost.

Primeri:

$$T(n) = 2T\left(\frac{n}{2}\right) + n$$

$$T(n) = T(n - 1) + n$$

$$T(n) = 2T(\sqrt{n}) + \lg n$$

Metoda stabla rekurzije

1. Formiramo stablo rekurzije (ispišemo složenost svih rekurzivnih poziva).
2. Sumiramo složenosti po nivoima rekurzije pa sumiramo sve nivoe.

Primeri:

$$T(n) = 3T\left(\frac{n}{4}\right) + n^2$$

$$T(n) = 2T(n - 1) + 1$$

$$T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + n$$

Master metoda

Rešava rekurencije oblika: $T(n) = aT\left(\frac{n}{b}\right) + f(n)$

Teorema: $a \geq 1, b > 1$:

1. $(\exists \varepsilon > 0) f(n) = O(n^{\log_b a - \varepsilon}) \rightarrow T(n) = \theta(n^{\log_b a})$

2. $f(n) = \theta(n^{\log_b a}) \rightarrow T(n) = \theta(n^{\log_b a} \lg n)$

3. $(\exists \varepsilon > 0) f(n) = \Omega(n^{\log_b a + \varepsilon}) \wedge (\exists c > 0)(\forall n > n_0) af\left(\frac{n}{b}\right) \leq cf(n) \rightarrow$
 $\rightarrow T(n) = \theta(f(n))$

Master metoda

Primeri:

$$T(n) = 9T\left(\frac{n}{3}\right) + n$$

$$T(n) = T\left(\frac{2n}{3}\right) + 1$$

$$T(n) = 3T\left(\frac{n}{4}\right) + n \lg n$$

$$T(n) = 2T\left(\frac{n}{2}\right) + n \lg n$$

$$T(n) = 7T\left(\frac{n}{2}\right) + n^2$$

Master metoda

Primeri:

$$T(n) = 9T\left(\frac{n}{3}\right) + n$$

$$T(n) = \theta(n^2)$$

$$T(n) = T\left(\frac{2n}{3}\right) + 1$$

$$T(n) = \theta(\lg n)$$

$$T(n) = 3T\left(\frac{n}{4}\right) + n \lg n$$

$$T(n) = \theta(n \lg n)$$

$$T(n) = 2T\left(\frac{n}{2}\right) + n \lg n$$

Nije primenjivo

$$T(n) = 7T\left(\frac{n}{2}\right) + n^2$$

$$T(n) = \theta(n^{\lg 7})$$



thank you!

© Universal Studios, Revealing Homes