Univerzitet Sinergija

Strukture podataka i algoritmi

leto 2011/12

Rekurzivni algoritmi

- Iterativni algoritam: ponavljanje nekog postupka (tela petlje) više puta
- Rekurzivni algoritam: ponavljanje nekog postupka (samog algoritma) više puta
- Rekurzivni algoritam je algoritam koji poziva sam sebe

Rekurzivni algoritmi

- Primer: x^n, x realan broj i $n \geq 0$ ceo broj
- Iterativno rešenje za x^n:

```
// Ulaz:  realan broj x, ceo broj n ≥ 0
// Izlaz: broj x^n
```

Rekurzivni algoritmi

- Primer: x^n, x realan broj i $n \geq 0$ ceo broj
- Iterativno rešenje za x^n:

```
// Ulaz:  realan broj x, ceo broj n ≥ 0
// Izlaz: broj x^n

algorithm power(x, n)

   y = 1;
   for i = 1 to n do
      y = x * y;

   return y;
```

Rekurzivni algoritmi

- Rekurzivno rešenje za x^n:

Rekurzivni algoritmi

- Rekurzivno rešenje za x^n:

$$
\begin{aligned}
x^n &= \underbrace{x \cdot x \cdot \dots \cdot x}_{n} \\
&= x \cdot \underbrace{x \cdot \dots \cdot x}_{n-1} \\
&= x \cdot x^{n-1}
\end{aligned}
$$

Rekurzivni algoritmi

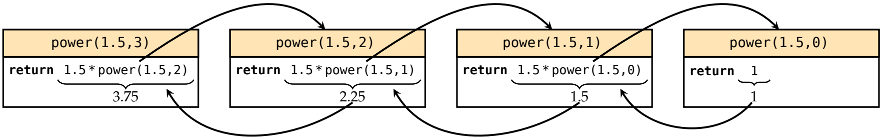
- Rekurzivno rešenje za x^n:

```
// Ulaz:  realan broj x, ceo broj n ≥ 0
// Izlaz: broj x^n
algorithm power(x, n)

   if (n == 0) then
      return 1;
   else
      return x * power(x,n-1);
```

Rekurzivni algoritmi

- Lanac rekurzivnih poziva istog algoritma
- Primer: `power(1.5,3)` → 3.75



Rekurzivni algoritmi

- Primer: $\mathrm{nzd}(x, y)$, $x \geq y > 0$ celi brojevi

Rekurzivni algoritmi

- Primer: $\mathrm{nzd}(x, y)$, $x \geq y > 0$ celi brojevi
- $d = \mathrm{nzd}(x, y)$ ako
  1. d je zajednički delilac za x i y (d deli oba broja x i y bez ostatka)
  2. d je <u>najveći</u> zajednički delilac za x i y

Rekurzivni algoritmi

- Primer: $\mathrm{nzd}(x, y)$, $x \geq y > 0$ celi brojevi
- $d = \mathrm{nzd}(x, y)$ ako
  1. d je zajednički delilac za x i y (d deli oba broja x i y bez ostatka)
  2. d je <u>najveći</u> zajednički delilac za x i y
- $\mathrm{nzd}(30, 12) = ?$

Rekurzivni algoritmi

- Primer: $\mathrm{nzd}(x,y)$, $x \geq y > 0$ celi brojevi
- $d = \mathrm{nzd}(x,y)$ ako
  1. d je zajednički delilac za x i y (d deli oba broja x i y bez ostatka)
  2. d je najveći zajednički delilac za x i y
- $\mathrm{nzd}(30,12) = ?$
  - delioci za 30: $\{1,2,3,5,6,10,15,30\}$
  - delioci za 12: $\{1,2,3,4,6,12\}$
  - zajednički delioci za 30 i 12: $\{1,2,3,6\}$
  - $\mathrm{nzd}(30,12) = 6$

Rekurzivni algoritmi

- Iterativno rešenje za $\mathrm{nzd}(x, y)$

Rekurzivni algoritmi

- Iterativno rešenje za $\mathrm{nzd}(x, y)$
- Ideja: počinjući od manjeg broja, proveravati sve manje brojeve dok se ne pronađe nzd

```
// Ulaz:  pozitivni celi brojevi x i y, x ≥ y
// Izlaz: nzd(x,y)
algorithm nzd(x, y)

   d = y;
   while ((x % d != 0) || (y % d != 0)) do
      d = d - 1;

   return d;
```

Rekurzivni algoritmi

- $\text{nzd}(12378, 3054) = ?$

Rekurzivni algoritmi

- $\mathrm{nzd}(12378, 3054) = ?$
- Brojevi koji se proveravaju da li su delioci oba broja 12378 i 3054:

$$3054, 3053, 3052, 3051, \ldots, 7, 6$$

- $\mathrm{nzd}(12378, 3054) = 6$

Rekurzivni algoritmi

- Rekurzivno rešenje za $\mathrm{nzd}(x, y)$
- $\mathrm{nzd}(12378, 3054) = ?$

Rekurzivni algoritmi

- Rekurzivno rešenje za $\mathrm{nzd}(x, y)$
- $\mathrm{nzd}(12378, 3054) = ?$

$12378, 3054$

Rekurzivni algoritmi

- Rekurzivno rešenje za $\mathrm{nzd}(x, y)$
- $\mathrm{nzd}(12378, 3054) = ?$

$12378, 3054$

$12378, 3054, 162 \qquad (12378 \,\%\, 3054 = 162)$

Rekurzivni algoritmi

- Rekurzivno rešenje za $\text{nzd}(x, y)$
- $\text{nzd}(12378, 3054) = ?$

$12378, 3054$

$12378, 3054, 162 \qquad (12378 \,\%\, 3054 = 162)$

$12378, 3054, 162, 138 \qquad (3054 \,\%\, 162 = 138)$

Rekurzivni algoritmi

- Rekurzivno rešenje za $\mathrm{nzd}(x, y)$
- $\mathrm{nzd}(12378, 3054) = ?$

$$12378, 3054$$
$$12378, 3054, 162 \qquad (12378 \,\%\, 3054 = 162)$$
$$12378, 3054, 162, 138 \qquad (3054 \,\%\, 162 = 138)$$
$$12378, 3054, 162, 138, 24 \qquad (162 \,\%\, 138 = 24)$$

Rekurzivni algoritmi

- Rekurzivno rešenje za $\mathrm{nzd}(x, y)$
- $\mathrm{nzd}(12378, 3054) = ?$

$12378, 3054$

$12378, 3054, 162 \qquad (12378 \,\%\, 3054 = 162)$

$12378, 3054, 162, 138 \qquad (3054 \,\%\, 162 = 138)$

$12378, 3054, 162, 138, 24 \qquad (162 \,\%\, 138 = 24)$

$12378, 3054, 162, 138, 24, 18 \qquad (138 \,\%\, 24 = 18)$

Rekurzivni algoritmi

- Rekurzivno rešenje za $\text{nzd}(x, y)$
- $\text{nzd}(12378, 3054) = ?$

$$
\begin{array}{ll}
12378, 3054 & \\
12378, 3054, 162 & (12378 \,\%\, 3054 = 162) \\
12378, 3054, 162, 138 & (3054 \,\%\, 162 = 138) \\
12378, 3054, 162, 138, 24 & (162 \,\%\, 138 = 24) \\
12378, 3054, 162, 138, 24, 18 & (138 \,\%\, 24 = 18) \\
12378, 3054, 162, 138, 24, 18, 6 & (24 \,\%\, 18 = 6)
\end{array}
$$

Rekurzivni algoritmi

- Rekurzivno rešenje za $\mathrm{nzd}(x, y)$
- $\mathrm{nzd}(12378, 3054) = ?$

$12378, 3054$

$12378, 3054, 162 \qquad (12378 \,\%\, 3054 = 162)$

$12378, 3054, 162, 138 \qquad (3054 \,\%\, 162 = 138)$

$12378, 3054, 162, 138, 24 \qquad (162 \,\%\, 138 = 24)$

$12378, 3054, 162, 138, 24, 18 \qquad (138 \,\%\, 24 = 18)$

$12378, 3054, 162, 138, 24, 18, 6 \qquad (24 \,\%\, 18 = 6)$

$12378, 3054, 162, 138, 24, 18, 6, 0 \qquad (18 \,\%\, 6 = 0)$

$\Rightarrow \mathrm{nzd}(12378, 3054) = 6$

Rekurzivni algoritmi

- Postupak nije beskonačan, jer se u sledećem koraku uvek dobija broj (ostatak) koji je striktno manji od prethodnog broja (delioca)
- Euklid:

  $x \geq y > 0$ celi brojevi $\Rightarrow \mathrm{nzd}(x, y) = \mathrm{nzd}(y, x\%y)$

Rekurzivni algoritmi

- Zašto zajednički delilac za x i y deli i $x\%y$ bez ostatka?
- x predstavljen pomoću rezultata deljenja i ostatka deljenja sa y:

$$x = y \cdot \lfloor x/y \rfloor + x\%y$$

- d bilo koji ceo broj koji deli x i y bez ostatka:

$$\frac{x}{d} = \frac{y \cdot \lfloor x/y \rfloor}{d} + \frac{x\%y}{d}$$

Rekurzivni algoritmi

- Rekurzivno rešenje za $\mathrm{nzd}(x, y)$:

$$\mathrm{nzd}(x, y) = \begin{cases} y, & \text{ako je } x \,\%\, y = 0 \\ \mathrm{nzd}(y, x \,\%\, y), & \text{ako je } x \,\%\, y \neq 0 \end{cases}$$

Rekurzivni algoritmi

- Euklidov algoritam

```
// Ulaz:  pozitivni celi brojevi x i y, x ≥ y
// Izlaz: nzd(x,y)
algorithm nzd(x, y)

   if (x % y == 0) then
      return y;
   else
      return nzd(y,x % y);
```

Rekurzivni algoritmi

- Rekurzivno rešavanje problema
  - Bazni slučaj
    - Rešenje trivijalno
  - Opšti slučaj
    - Podeliti dati problem u više (nezavisnih) sličnih manjih problema
    - Rešiti manje probleme rekurzivno
    - Sastaviti rešenja manjih problema u rešenje datog polaznog problema

Rekurzivni algoritmi

- Primer: najveći element niza

```
// Ulaz:  niz a, njegov broj elemenata n
// Izlaz: indeks najvećeg elementa niza a
algorithm max(a, n)
```

Rekurzivni algoritmi

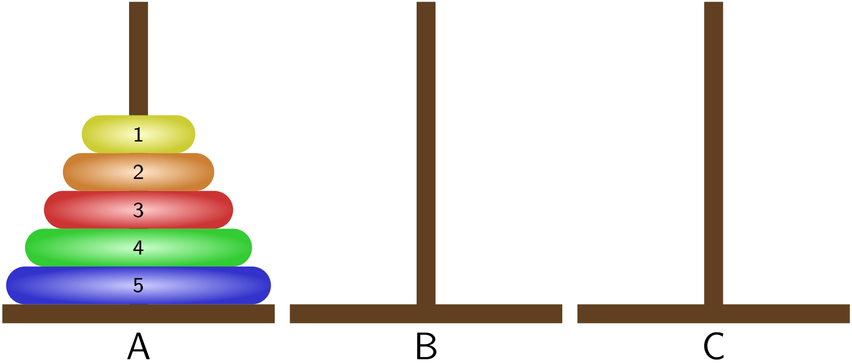
- Primer: najveći element niza

```
// Ulaz:  niz a, njegov broj elemenata n
// Izlaz: indeks najvećeg elementa niza a
algorithm max(a, n)

   if (n == 1) then // bazni slučaj
      return 1;
   else             // opšti slučaj
      i = max(a, n-1);
      if (a[i] < a[n]) then
         return n;
      else
         return i;
```

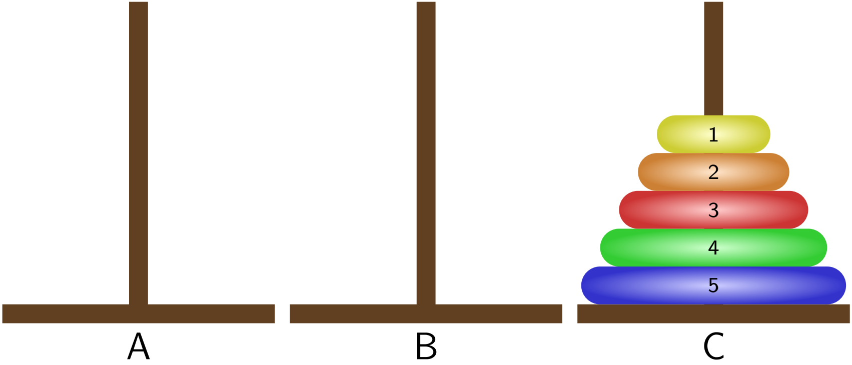
Rekurzivni algoritmi

- Primer: igra Hanojske kule



Rekurzivni algoritmi

- Cilj igre Hanojske kule:



Rekurzivni algoritmi

- Cilj igre: premestiti sve diskove sa početnog stuba A na ciljni stub C tako da se diskovi nalaze u istom poretku, koristeći srednji stub B kao pomoćni
- Jedan potez za premeštanje diskova:
  1. Samo se disk sa vrha nekog stuba može skinuti i staviti na vrh drugog stuba
  2. Nikad se veći disk ne može staviti iznad manjeg diska

Rekurzivni algoritmi

- Primeri toka igre Hanojskih kula za 1, 2, 3, 4 diska

Rekurzivni algoritmi

- Ideja za rekurzivno rešenje Hanojskih kula sa n diskova:
  - Pretpostavimo da imamo rešenje manjeg problema Hanojskih kula za premeštanje $n-1$ diska sa jednog stuba na drugi stub koristeći treći stub kao pomoćni
  - Kako to rešenje manjeg problema iskoristiti da bi se konstruisalo rešenje za polazni problem Hanojskih kula sa n diskova?

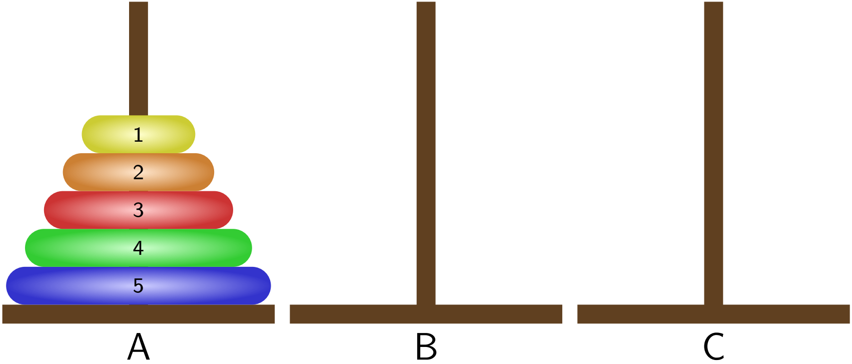
Rekurzivni algoritmi

- Rekurzivni algoritam za Hanojske kule sa n diskova:

  1. Rekurzivno premestiti $n-1$ gornjih diskova sa stuba A na stub B koristeći stub C kao pomoćni
  2. Premestiti najveći disk sa stuba A na stub C
  3. Rekurzivno premestiti $n-1$ diskova sa stuba B na stub C koristeći stub A kao pomoćni

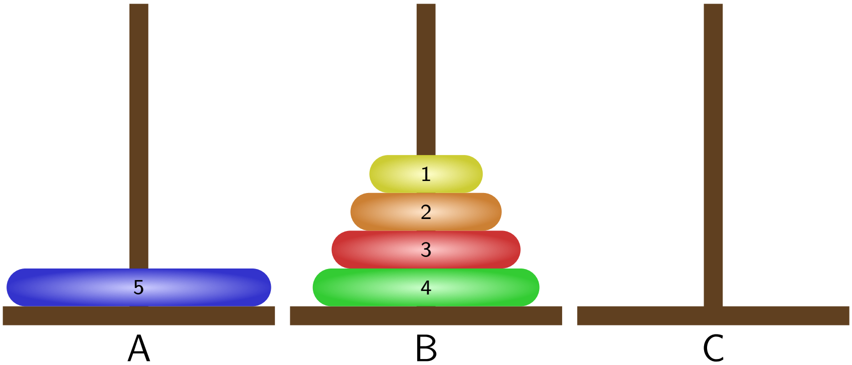
Rekurzivni algoritmi

- Početna konfiguracija



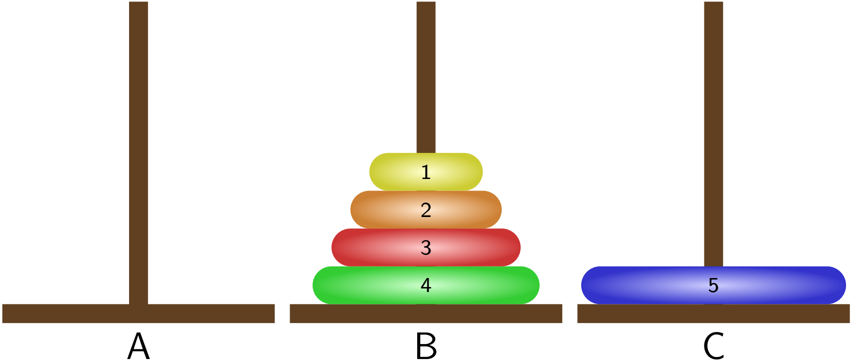
Rekurzivni algoritmi

- Rekurzivno premestiti $n-1$ gornjih diskova sa stuba A na stub B koristeći stub C kao pomoćni



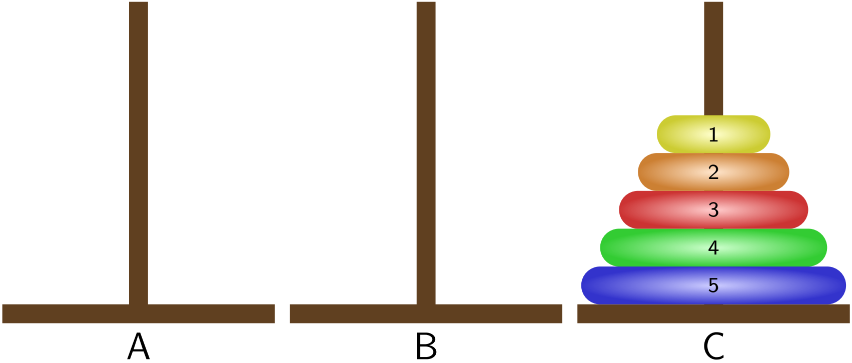
Rekurzivni algoritmi

- Premestiti najveći disk sa stuba A na stub C



Rekurzivni algoritmi

- Rekurzivno premestiti $n - 1$ diskova sa stuba B na stub C koristeći stub A kao pomoćni



Rekurzivni algoritmi

- Rekurzivni algoritam za Hanojske kule sa n diskova

```
// Ulaz: broj diskova, početni, pomoćni i ciljni stub
// Izlaz: niz poteza za Hanojske kule sa n diskova
algorithm toh(n, a, b, c)

   if (n == 1) then  // bazni slučaj
      move(a,c);
   else              // opšti slučaj
      toh(n-1,a,c,b);
      move(a,c);
      toh(n-1,b,a,c);

   return;
```