Rekurzije

ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

Ova publikacija je izrađena uz pomoć Europske unije. Sadržaj publikacije je u isključivoj nadležnosti V. gimnazije te se ni na koji način ne može smatrati da odražava stajališta Europske unije.

Europsku uniju čini 28 država članica koje su odlučile postupno povezivati svoja znanja, resurse i sudbine. Tijekom 50-godišnjeg razdoblja proširivanja, zajedno su izgradile područje stabilnosti, demokracije i održivog razvoja, zadržavajući pritom kulturnu raznolikost, toleranciju i slobode pojedinaca. Europska unija je posvećena dijeljenju svojih postignuća i vrijednosti s državama i narodima izvan svojih granica.

Pristupi rješavanju problema

- **induktivni pristup**
  - od manjih objekata i dimenzija
- **rekurzivni pristup**
  - od većih dimenzija prema manjima – za rješenje se koriste rješenja drugih stanja

ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 1

- Napišimo funkciju čiji će parametar biti prirodan broj *n* a vraćat će zbroj prvih n prirodnih brojeva.

ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 1 – induktivno rješenje

- brojeve počinjemo zbrajati od 1 dodajući redom brojeve do *n*

```
def zbroj_i(n):
    z = 0
    for i in range(1, n + 1):
        z += i
    return z
```

ULAGANJE U BUDUĆNOST
PROJEKT SUFINANCIRA EUROPSKA UNIJA
STRUKTURNI I INVESTICIJSKI FONDOVI
ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 1 – rekurzivno rješenje

- pretpostavljamo da znamo koliki je zbroj svih brojeva do 1, 2,… n – 1
- za rješenje koristimo neka od „poznatih" rješenja
- pretpostavimo da nas zanima zbroj prvih 5 prirodnih brojeva: s(5)
  - poznato nam je s(1), s(2),…, s(4)
  - zbroj prvih 5 brojeva jednak je zbroju prva 4 broja (s(4)) + 5 : s(5) = s(4) + 5
- općenito: s(n) = s(n – 1) + n – **rekurzivna relacija**
- **uvjet prekida**: zbroj prvih 1 prirodnih brojeva je 1

```
def zbroj_r(n):
    if n == 1:
        return 1
    else:
        return zbroj_r(n - 1) + n
```

ULAGANJE U BUDUĆNOST
PROJEKT SUFINANCIRA EUROPSKA UNIJA
STRUKTURNI I INVESTICIJSKI FONDOVI
ESF RAZVOJ LJUDSKIH POTENCIJALA

Rekurzivne funkcije

- rekurzivna relacija – odnos traženog rješenja s obzirom na neko poznato rješenje
- uvjet prekida – rješenje za neku trivijalnu dimenziju

ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Izvođenje rekurzivne funkcije

Iz rekurzivne relacije f(n) = f(n – 1) + n proizlazi da je f(5) = f(4) + 5.
Čim se pojavi f(4) dolazimo do računanje vrijednosti f(4) (f(4) = f(3) + 4).
Ovim postupkom dolazimo do f(1) za koji (iz rekurzivne relacije) znamo da je jednak 1.
Budući da je f(1) jednak 1 slijedi da je f(2) = 3, f(3) = 6, f(4) = 10 te na kraju je f(5) jednak 15, što je traženo rješenje.
Za pohranjivanje međurezultata i izraza u ovom slučaju se koristi struktura stoga.
Primijetimo da ukoliko ne bi bilo uvjeta prekida (f(1) = 1), rekurzivna funkcija bi se nastavila izvoditi

f(5) = f(4) + 5

f(4) = f(3) + 4

f(3) = f(2) + 3

f(2) = f(1) + 2

f(1) = 1

ULAGANJE U BUDUĆNOST
PROJEKT SUFINANCIRA EUROPSKA UNIJA
STRUKTURNI I INVESTICIJSKI FONDOVI
ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 2

- Napišimo rekurzivno funkciju koja će ispisivati rastav prirodnog broja *n* na proste faktore.

ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 2 – rješenje

- **r(n)** – vraća rastav broja *n* na faktore
- poznat je **r(i)** za svaki **i ≠ n**
- tražimo odnos **r(n)** i nekog **r(i)** – želimo pronaći odnos rastava broja n na proste faktore ako znamo rastaviti neki drugi broj na proste faktore:
  - ako je n djeljiv s 2: **r(n) = ‘ * 2’ + r(n / 2)**
  - ako n nije djeljiv s 2:
    - ako je n djeljiv s 3:
      - **r(n) = ‘ * 3’ + r(n / 3)**
      - …

ULAGANJE U BUDUĆNOST
PROJEKT SUFINANCIRA EUROPSKA UNIJA
STRUKTURNI I INVESTICIJSKI FONDOVI
ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 2 – rješenje

- **r(n, k)** – rastav broja n na proste faktore pri čemu je k najmanji potencijalni djelitelj broja n
  - r(n, k) = ‘k * ‘ + r(n / k, k) – ako je n djeljiv s k
  - r(n, k + 1) – ako n nije djeljiv s k
- uvjet prekida – ukoliko je n = k – budući da je k potencijalni djelitelj broja n nema smisla promatrati djelitelje veće od k

ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 2 – rješenje

```
def r(n, k):
    if n == k:
        return str(k)
    elif n % k == 0:
        return str(k) + ' * ' + r(n // k, k)
    else:
        return r(n, k + 1)
```

ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 3

- Napišimo rekurzivno funkciju koja će vraćati n-ti po redu element Fibonaccijevog niza.

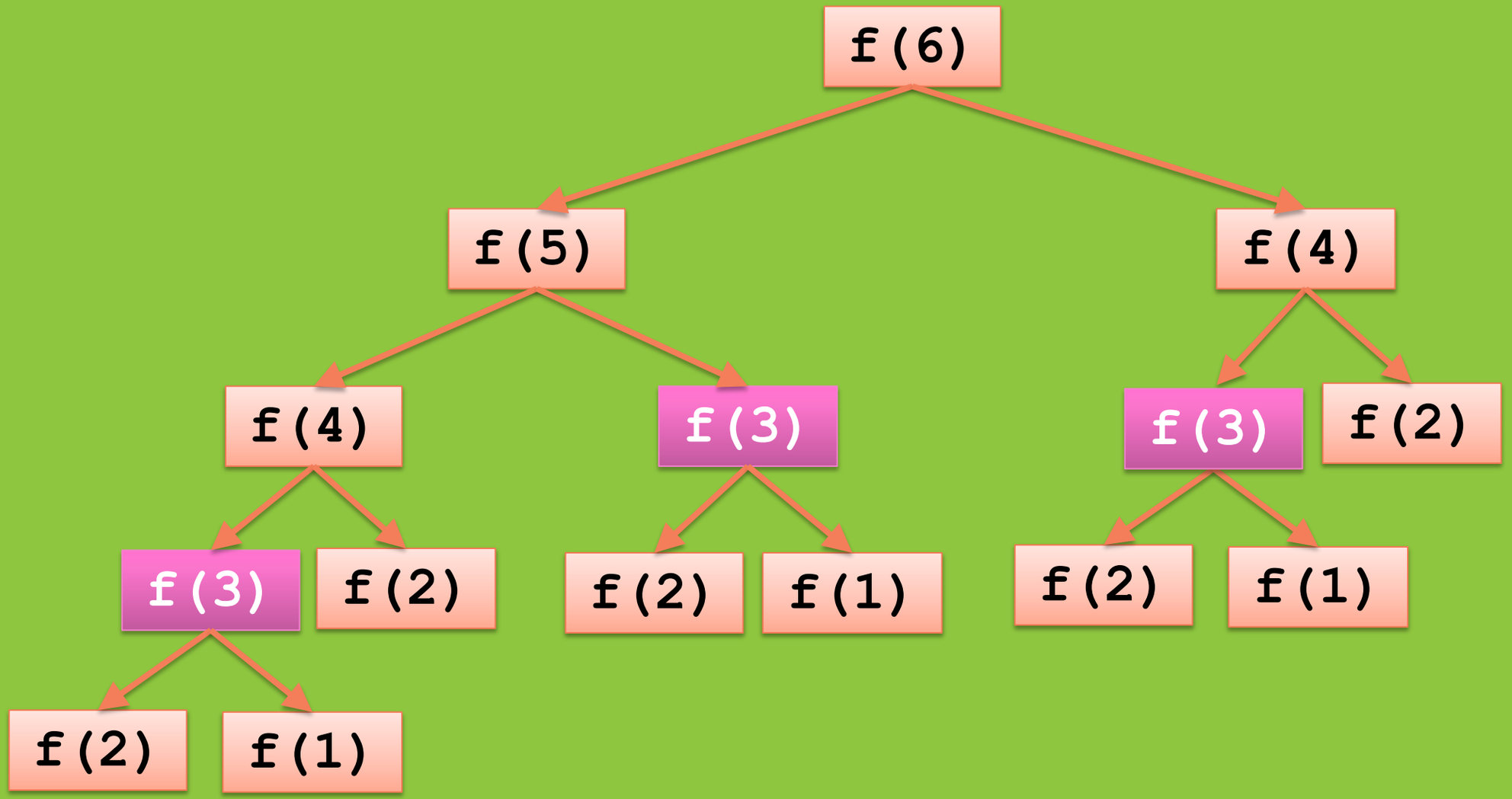
- rekurzivna relacija: `f(n) = f(n - 1) + f(n - 2)`
- uvjet prekida: `f(1) = 1, f(2) = 1`

Primjer 3 - rješenje

- 1, 1, 2, 3, 5, 8, 13,...

```
def fib(n):
    if n < 3:
        return 1
    else:
        return fib(n - 1) + fib(n - 2)
```

Primjer 3 - ilustracija



ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi

- Na prvom od 3 štapa nalazi se n diskova poredanih od najvećeg prema najmanjem. Potrebno je preseliti sve diskove s prvog na drugi štap, pri čemu je treći štap pomoćni a treba poštovati sljedeća pravila:
  - u svakom koraku može se preseliti najviše jedan disk
  - ni u kom trenutku veći disk se ne može nalaziti na manjem disku

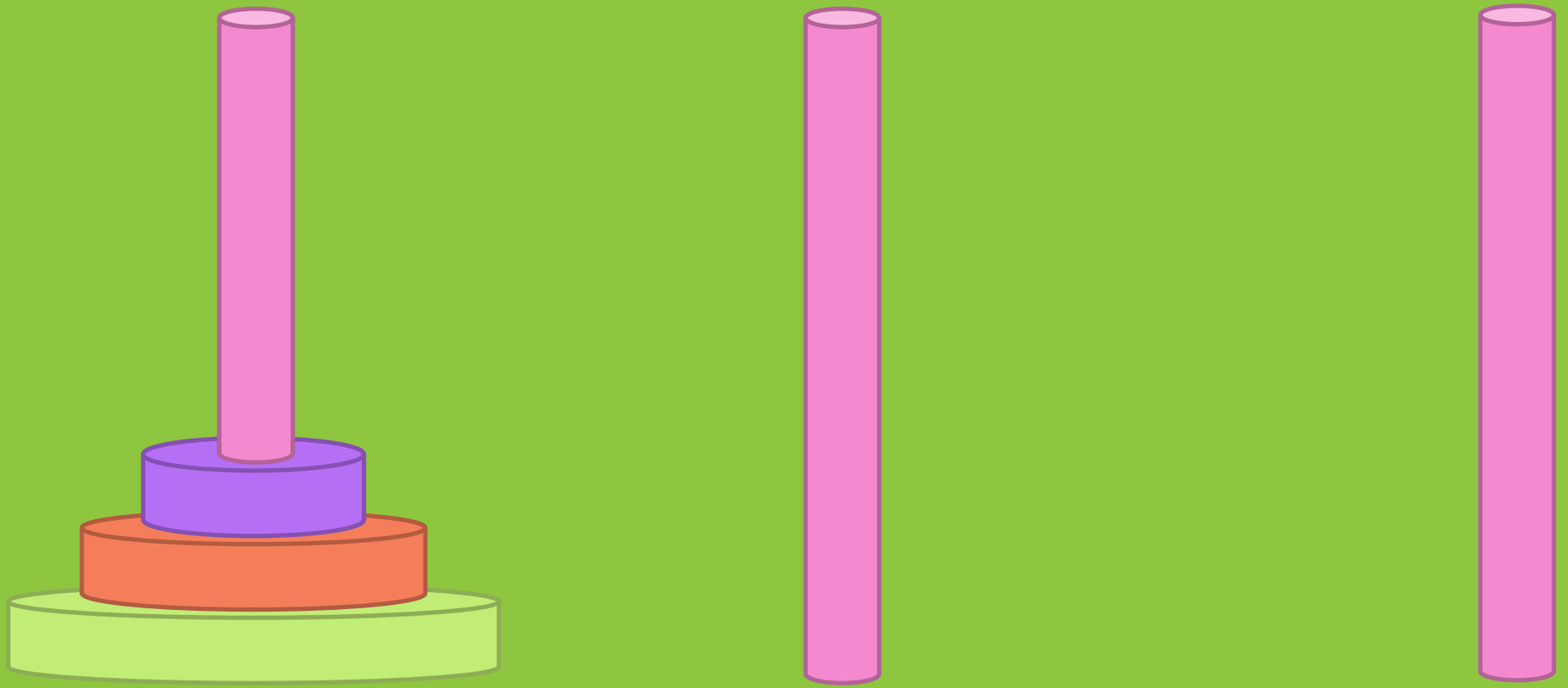
ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi - ilustracija



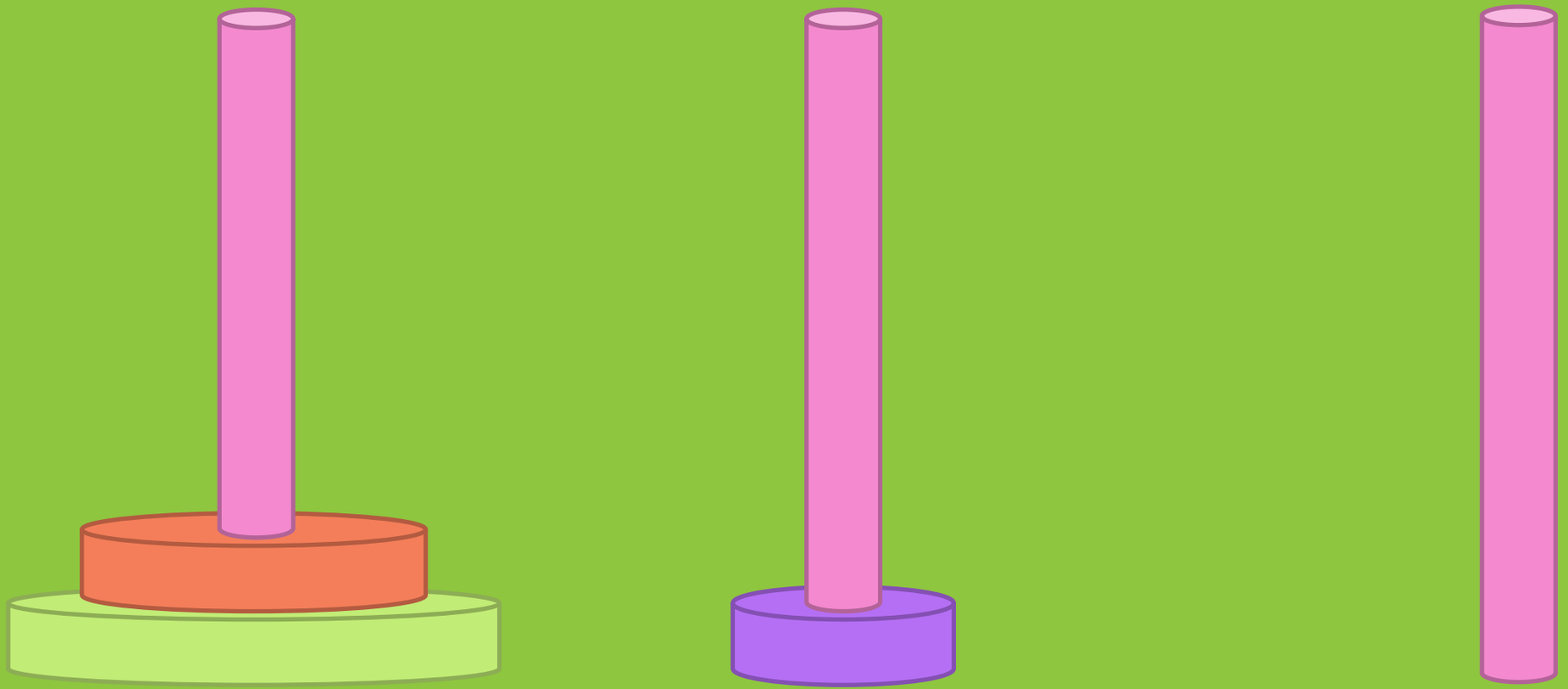
ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi - ilustracija



ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi - ilustracija

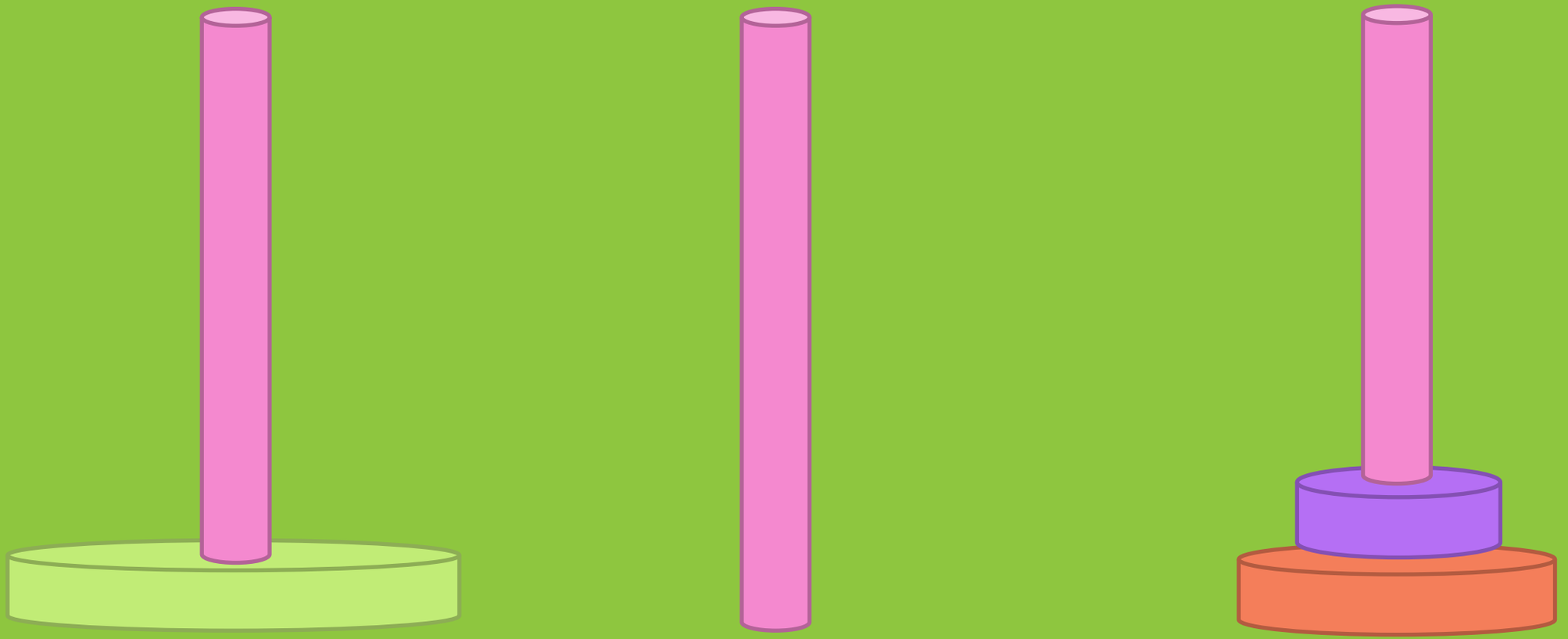
ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi - ilustracija



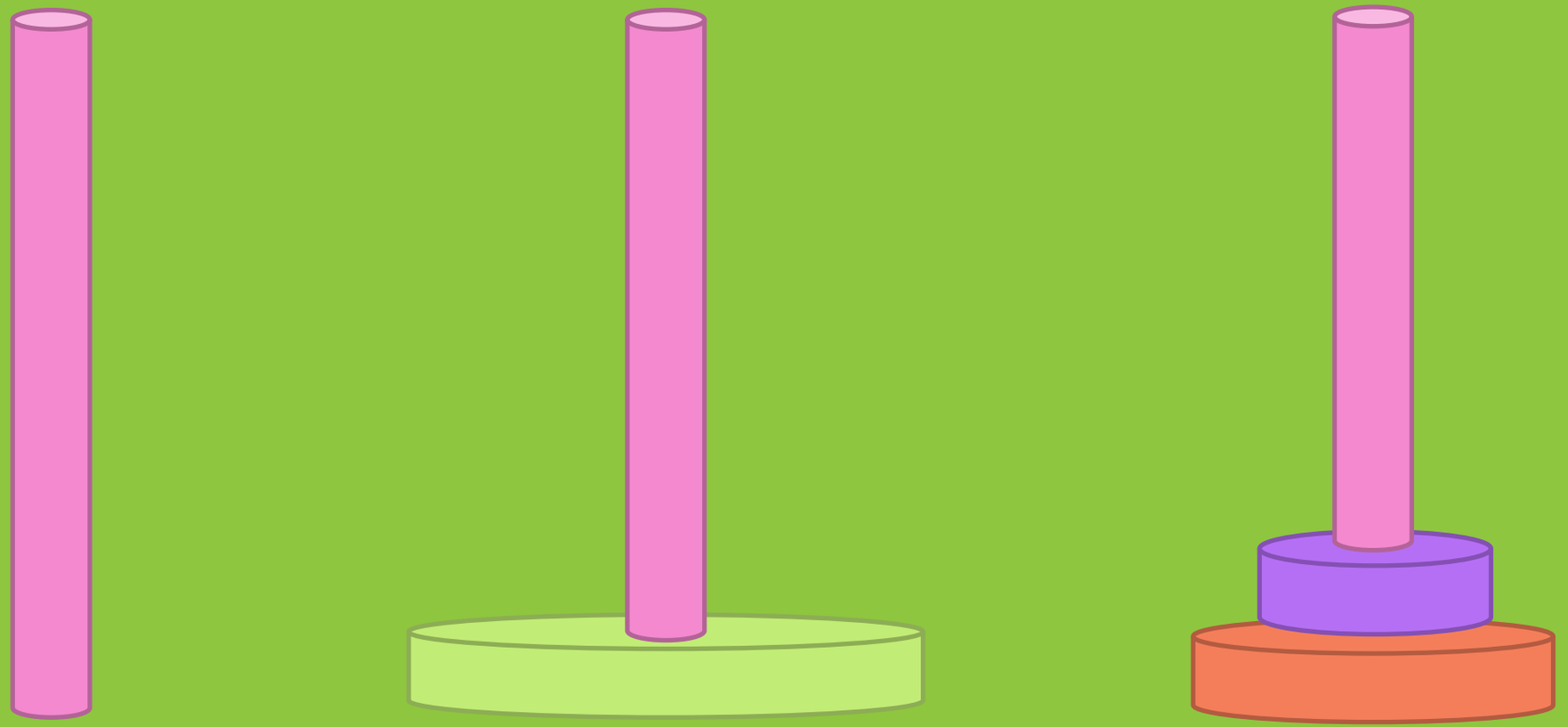
ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi - ilustracija



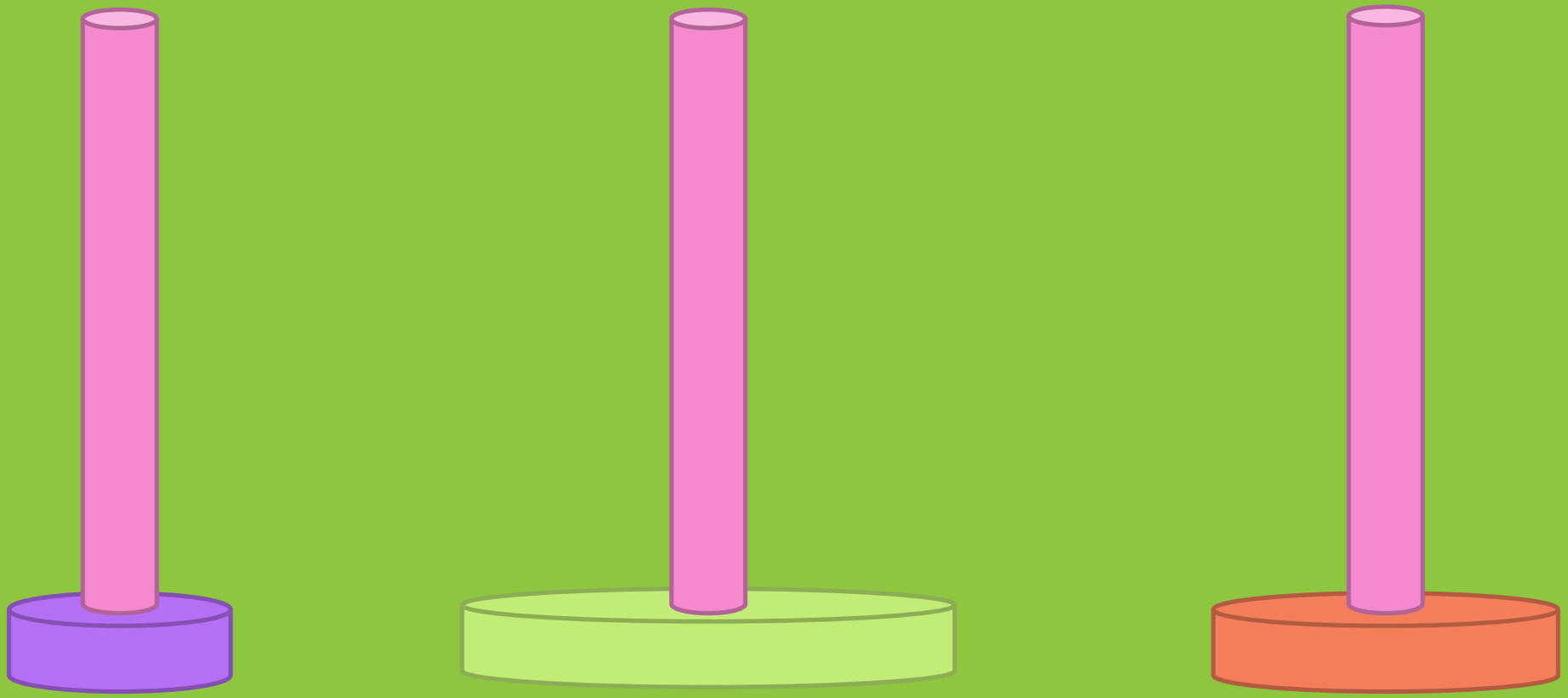
ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi - ilustracija



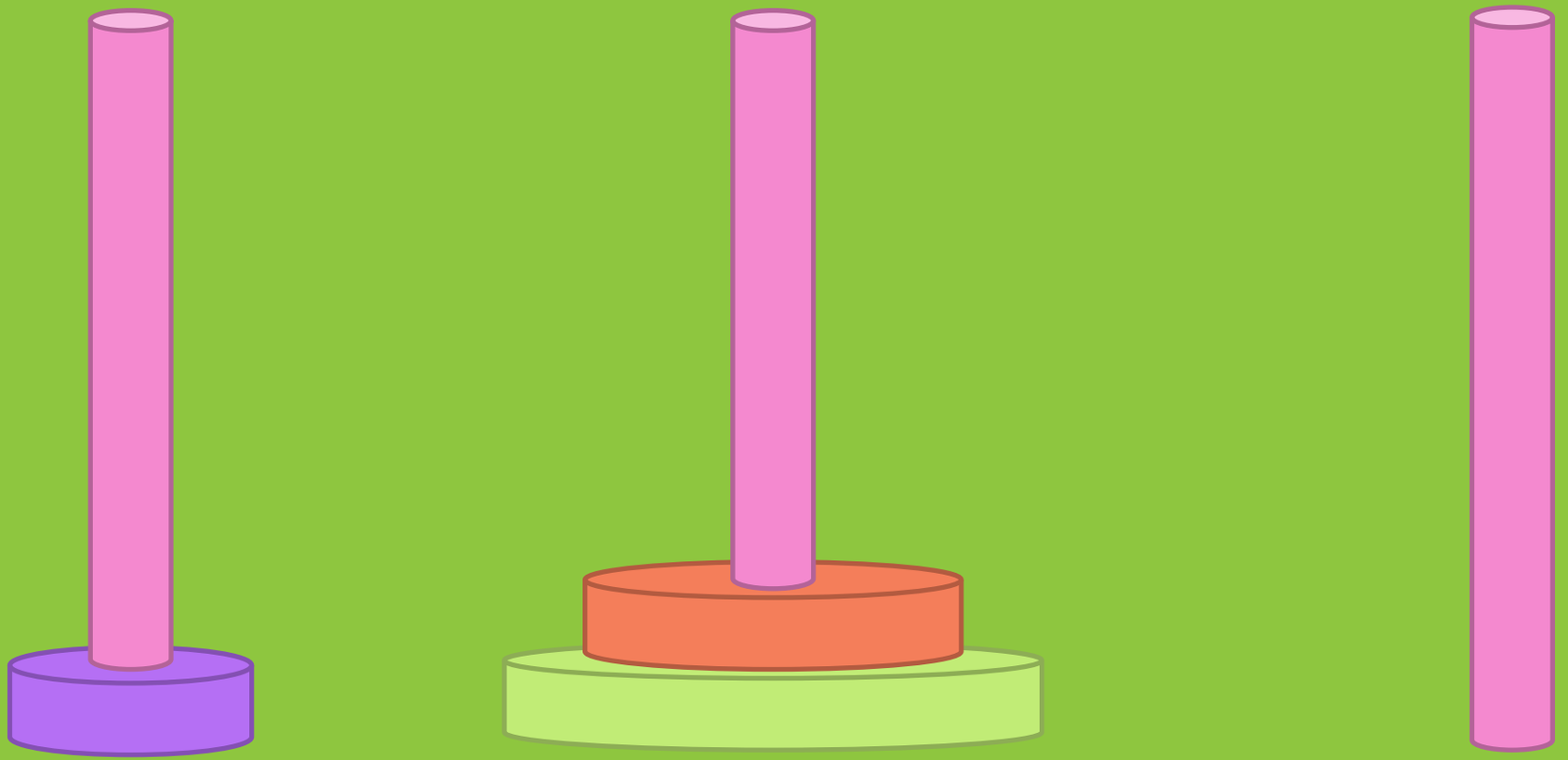
ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi - ilustracija



ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi - ilustracija

ULAGANJE U BUDUĆNOST

PROJEKT
SUFINANCIRA
EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI
FONDOVI

ESF
RAZVOJ
LJUDSKIH
POTENCIJALA

Primjer 4 – Hanojski tornjevi - rješenje

- tražimo postupak prebacivanja *n* diskova s prvog na drugi štap pri čemu je treći štap pomoćni
- „znamo" prebaciti bilo koji drugi broj (osim n) diskova s bilo kojeg štapa na bilo koji drugi štap
- znamo prebaciti n – 1 diskova s prvog na treći štap
- rekurzivna relacija:
  - prebaciti n – 1 diskova s prvog na treći štap
  - prebaciti disk s prvog na drugi štap
  - prebaciti n – 1 diskova s trećeg na drugi štap
- uvjet prekida – ukoliko imamo samo jedan disk prebacimo ga s prvog na drugi štap

ULAGANJE U BUDUĆNOST
PROJEKT SUFINANCIRA EUROPSKA UNIJA
STRUKTURNI I INVESTICIJSKI FONDOVI
ESF RAZVOJ LJUDSKIH POTENCIJALA

Primjer 4 – Hanojski tornjevi - rješenje

```
def hanoi(n, a = 'A', b = 'B', c = 'C'):
    if n == 1:
        print('s {} na {}'.format(a, b))
    else:
        hanoi(n - 1, a, c, b)
        print('s {} na {}'.format(a, b))
        hanoi(n - 1, c, b, a)
        return
```

ULAGANJE U BUDUĆNOST

PROJEKT SUFINANCIRA EUROPSKA UNIJA

STRUKTURNI I INVESTICIJSKI FONDOVI

ESF RAZVOJ LJUDSKIH POTENCIJALA