

Podatkovne strukture in algoritmi (2024/25)

1. domača naloga – teoretični del

1. naloga:

- (i) Naslednje funkcije razvrstite v nepadajočem vrstem redu glede na asimptotično rast (asimptotično kompleksnost veliki \mathcal{O}):

$$f_1(n) = 2^{2^{1000000}}, f_2(n) = 2^{1000000n}, f_3(n) = \binom{n}{2}, f_4(n) = n\sqrt{n}.$$

Vse odgovore je potrebno utemeljiti!

Primer: Funkcija $f(n) = n$ raste asimptotično počasneje kot funkcija $g(n) = n^2$; z drugimi besedami, $f(n) \in \mathcal{O}(g(n))$, toda $g(n) \notin \mathcal{O}(f(n))$.

- (ii) Z uporabo definicije velikega \mathcal{O} pokažite $n^{1+0,001} \notin \mathcal{O}(n)$.

2. naloga: Naj bosta f in g pozitivni funkciji. So naslednje trditve pravilne ali nepravilne? Utemeljite.

- (i) $f(n) = \mathcal{O}(g(n)) \Leftrightarrow 2^{f(n)} = \Omega(2^{g(n)})$;
(ii) $f(n) = \Theta(f(\frac{n}{2}))$;
(iii) $f(n) + g(n) = \theta(\min\{f(n), g(n)\})$, kjer

$$\min\{f(n), g(n)\}(n) = \begin{cases} f(n); & \text{če } f(n) \leq g(n) \\ g(n); & \text{sicer.} \end{cases}$$

3. naloga: Dan je naslednji problem:

Vhod: Zaporedje števil $A = [a_1, a_2, \dots, a_n]$ in vrednost v .

Izhod: Indeks i , če je $v = A[i]$, ali poseben znak NIL, če v ni vsebovan v A .

- (i) Napišite algoritem za linearno iskanje, ki se sprehodi čez zaporedje in išče v .
(ii) Z uporabo zanke invariante pokažite pravilnost algoritma.

4. naloga: Naj bo A polje celih števil dolžine n . Dan je naslednji algoritem:

```
foo(A)
  n ← A.length
  x ← A[1]
  for i ← 2 to n do
    if A[i] > x then
      x = A[i]
  return x
```

- (i) Kaj vrne klic $foo(A)$?
- (ii) Napišite zanko invarianto za $foo(A)$.
- (iii) Z uporabo zanke invariante pokažite pravilnost algoritma $foo(A)$.

5. naloga: Dan je naslednji algoritem:

```
int foo(int x, y)
  if x == 0 return y
  y = 2 · y + x%2
  return foo(x/2, y)
```

- (i) Izračunajte $foo(5, 0)$, $foo(6, 0)$, $foo(7, 0)$.
- (ii) Kakšna je časovna zahtevnost klica $foo(n, 0)$?
- (iii) Naj bo $f_1(n) = 12n + 11\log(n)$ in $f_2(n) = \frac{11}{n} + 12\log(n)$. Za vsako funkcijo določite, kateri izmed naslednjih družin pripada:

$$O(n), \Omega(n), \Theta(n), O(\log n), \Omega(\log n), \Theta(\log n).$$

6. naloga: Dan je naslednji algoritem:

```
int FooBar(c, n):
  int p = 2 ★★ n
  result = 1
  while (p > 0):
    result = result ★ c
    p = p - 1
  return result
```

- (i) Kakšna je časovna zahtevnost funkcije *FooBar* v odvisnosti od n ? Utemeljite odgovor.
- (ii) Ali lahko pospešite funkcijo *FooBar*? Če ja, kako in kakšna je časovna zahtevnost, in če ne, zakaj ne.
- (iii) Pokažite:

$$n \ln n = O(n^{3/2})$$

in

$$n \ln n = \Omega(n^{1/2}).$$

Rešitev nalog oddajte preko e-učilnice: **oddajte .pdf datoteko**. Vse naloge je potrebno reševati **samostojno**. Prepisovanje se kaznuje z negativnimi točkami. Rok za oddajo nalog je **nedelja, 10. november 2024**.