

Vzporedni in porazdeljeni sistemi in algoritmi: Vaje

OpenMP

PREDAVATELJ: PATRICIO BULIĆ

ASISTENT: DAVOR SLUGA, ROK ČEŠNOVAR

Vaja 5

Števili sta prijateljski, če je vsota deljiteljev prvega števila enaka drugemu številu in obratno.

- http://en.wikipedia.org/wiki/Amicable_numbers
- <http://www.shyamsundergupta.com/amicable.htm>

Primer: 220 in 284 sta prijateljski števili

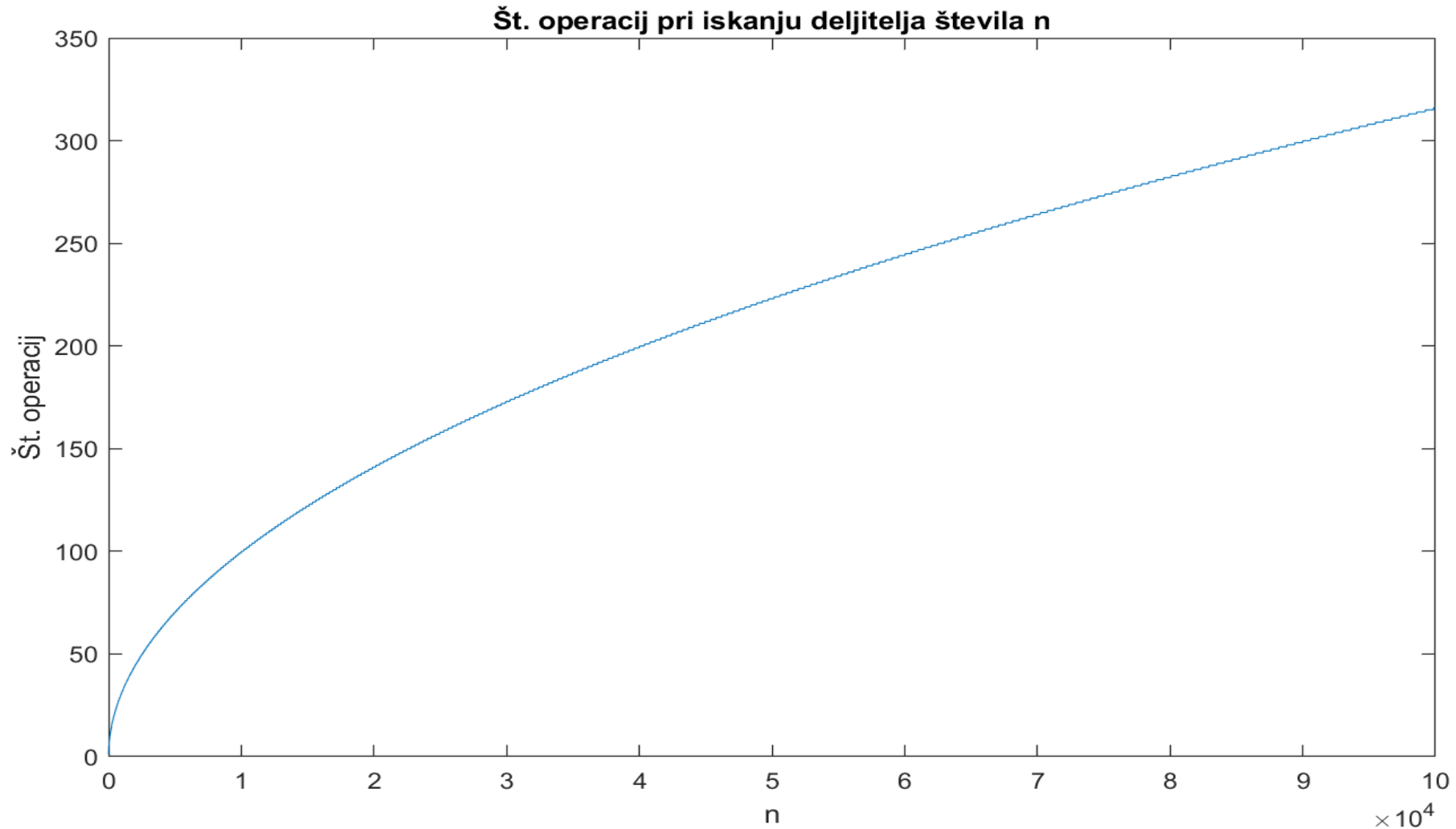
- Deljitelji 220: {1, 2, 4, 5, 10, 11, 20, 22, 44, 55, 110}
 - $1 + 2 + 4 + 5 + 10 + 11 + 20 + 22 + 44 + 55 + 110 = 284$
- Deljitelji 284: {1, 2, 4, 71, 142}
 - $1 + 2 + 4 + 71 + 142 = 220$

Vaja 5

Napišite vzporeden program s pomočjo OpenMP, ki bo izračunal vsoto vseh parov prijateljskih števil na intervalu $[1, N]$, kjer je N poljubno število.

Katere dele algoritma se splača paralelizirati?

Vaja 5



Učinkovita delitev
dela med niti?

Vaja 5

Delo enakomerno porazdelite med niti. Uporabite različne pristope za delitev dela med niti

- Dopolnilo schedule pri paralelizaciji zanke for
 - static
 - dynamic
 - guided
- Katera delitev je najbolj primerna (najhitrejše izvajanje)?

Vaja 5

Izmerite čas izvajanja na enem jedru in več jedrih ter primerjajte rezultate (1, 2, 4, 8, 16, 32). N mora biti dovolj velik!

Primerjate čase izvajanja za različne načine razporejanja dela med niti

Kolikšno pohitritev ste dosegli za vsak primer?

- Pohitritev $S = T_s/T_p$
- T_s -> čas sekvenčnega algoritma
- T_p -> čas paralelnega algoritma

Rezultate zabeležite na koncu vaše kode kot komentar.

Namigi

Pazite na globalne spremenljivke, ki si jih delijo vse niti

- uporaba kritičnih sekcij/atomičnih operacij/redukcije

Pazite, da posameznih prijateljskih števil ne štejete večkrat v vsoti

Čas izvajanje merite pri vključenih optimizacijah programske kode

- Stikalo prevajalnika gcc -O2

OpenMP omogočimo s stikalo -fopenmp, ko prevajamo

Rezervacija resursov na NSC: `salloc -n1 --cpus-per-task=<št. niti> -`
`-reservation=fri`

Vaja 5

Rok za oddajo: 25. 11. 2021; zadnji možni: 2. 12. 2021