

# Vizualizacija Knutovog algoritma

## 1. Opis problema

Da bismo brojali koliko puta se neki događaj desio prilikom izvršavanja programa, moramo da ubacimo kod koji će vršiti uvećanje brojača. Ovo uvodi određene troškove. Najjednostavniji način bi bio da za svaki blok ili ivicu ubacimo brojač. Tada bismo imali neophodne podatke, ali bi oni zauzeli previše računarskih resursa. Cilj nam je da minimizujemo broj umetnutih brojača. Broj izvršavanja svakog bloka možemo dobiti pomoću brojača ivica, tako što sumiramo sve ivice koje ulaze u taj blok. Profajliranje ivica može da se implementira značajno efikasnije, pod pretpostavkom da se radi u fazi kompilacije programa. Jedno takvo rešenje prvi je teoretski uveo Donald Knut (eng. *Donald Knuth*), dobitnik Tjuringove nagrade 1974. godine, i on je pokazao da je broj umetnutih brojača u njegovom rešenju minimalan [[1](#)].

## 2. Opis arhitekture

Projekat je rađen u programskom jeziku *Python*.

Korisceni su paketi i moduli:

- *Tkinter*
- *Unittest*

## 3. Opis rešenja problema

Ideja projekta je poboljšanje vizualizacije i implementacije Knutovog algoritma koji je rađen u projektu [[2](#)]. U odnosu na pomenuti projekat, dodata je mogućnost unosa novog koda za vizualizaciju algoritma. Takođe, dodati su novi test primeri koji se mogu izabrati klikom na ime test fajla na početku rada algoritma. Poboljšano je iscrtavanje grafa, ispisivanje grešaka za nevalidne akcije i dodata mogućnost inkrementiranja težina grana inverznog grafa. Sve funkcionalnosti algoritma su testirane korišćenjem paketa "*unittest*".

