

# Nelson-Openov algoritam za određivanje kongruentnog zatvorenja

## Uputstvo za pokretanje programa

- Nakon pokretanja terminala i pozicioniranja u direktorijumu, pokretanjem komande **make** generiše se izvršni program **main**. Nakon pokretanja komande **./main**, na ulazu treba uneti početne parametre.
- Prvi parametar je domen, u formatu:

$D = \{\text{niz\_termova}\};$

Simboli promenljivih označavaju se malim slovom abecede, a funkcijski simboli velikim slovima. Primer:

$D = \{x, y, F(x), G(y)\};$

- Drugi parametar je niz baznih jednakosti, u formatu:

$E = \{\text{niz\_parova}\};$

Par se zapisuje u formatu **term=term**, dok su parovi međusobno odvojeni zapetom. Primer:

$E = \{x=y, y=z\};$

- Primer ispravnog ulaza nalazi se u fajlu **input**, pa se program nakon kompilacije može pokrenuti naredbom **./main <input**.

## O algoritmu

Nelson-Openov algoritam na osnovu domena i niza baznih jednakosti nalazi klase kongruencije. Algoritam koristi pomoćne funkcije:

- **use(Term t)** – vraća vektor termova čiji je t podterm;
- **find(Term t)** – vraća kanonskog predstavnika klase kojoj pripada t;
- **union(Term s, Term t)** – spaja klase kojoj pripadaju termovi s i t;
- **cong(Term s, Term t)** – ako je term s oblika  $f(s_1, \dots, s_n)$  i t oblika  $f(t_1, \dots, t_n)$  i za svako  $t_i, s_i$  važi  $\text{find}(t_i) = \text{find}(s_i)$ , vraća **true**, inače **false**;

## *Pseudo-kod Nelson-Openovog algoritma*

```
function cc(E,T)
begin
    foreach t : T find(t) = t
    foreach e : E
        medge(e.first,e.second)
end
function merge(s,t)
begin
    S = U{use(u) | find(u) = find(s)}
    T = U{use(u) | find(u) = find(t)}
    union(s,t)
    foreach u1 : S
        foreach u2 : T
            if(find(u1)  $\neq$  find(u2) and cong(u1,u2))
                merge(u1,u2)
end
```