

Strukturální testování - datový tok

Skupina: 10

Řešitelé: Stanke Michal, Tímr Marek, Voříšek Lukáš

Zadání úlohy

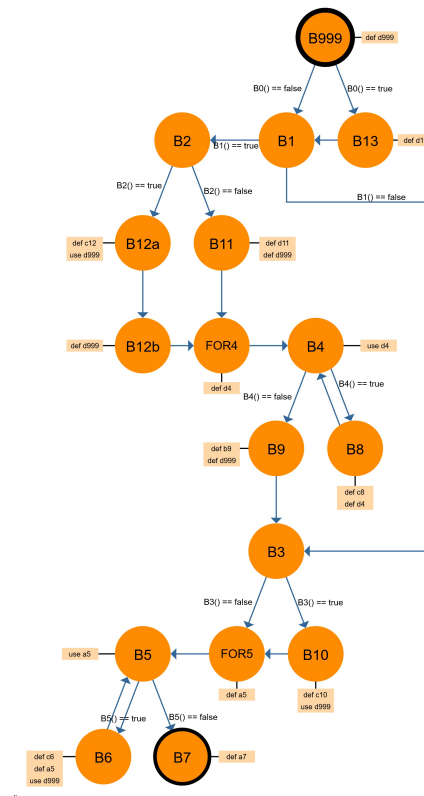
Následující zdrojový Java kód funkce převeďte na graf řídicího toku (převáděno na datový tok). Nalezněte du-cesty pro vybranou proměnnou ?999.

B999	public int fnc() { B999(); int d999 = 360; if(B0()) then {
B13	B13(); int c13 = 299; }
B1	if(B1()) then {
B2	if(B2()) then {
B12a	B12(); int c12 = d999;
B12b	d999 = 206; }
B11	else { B11(); int d11 = 996; d999 = 133; }
FOR4	for(int d4 = 0; B4(d4); d4++) {
B4	for(int d4 = 0; B4(d4); d4++) {
B8	for(int d4 = 0; B4(d4); d4++) { B8(); int c8 = 837; }
B9	B9(); int b9 = 814; d999 = 42; }
B3	if(B3()) then {
B10	B10(); int c10 = d999; }
FOR5	for(int a5 = 0; B5(a5); a5++) {
B5	for(int a5 = 0; B5(a5); a5++) {
B6	for(int a5 = 0; B5(a5); a5++) { B6(); int c6 = d999; }
B7	B7(); int a7 = 697; }

Převod na graf

Úryvek kódu ze zadání převedeme na graf a to tak, že uzly budou reprezentovat *vrcholy* grafu a řídicí funkce (respektive jejich vyhodnocení) bude představovat hrany grafu.

Pro každý vrchol ještě vedeme záznam definovaných proměnných (klíčové slovo *def*) a proměnných použitých (klíčové slovo *use*).



obrázek 1 - Model datového toku

du-cesty pro proměnnou $d999$

Du-cesta je vzhledem k proměnné $d999$ jednoduchou cestou, která je def-čistá z uzlu n_i do uzlu n_j

Cesty $du(n_i, d999)$

Jedná se o cesty vzhledem k proměnné $d999$, které začínají v uzlu n_i .

Označení cesty	Cesta
$du(b999, d999)$	b999, b1, b2, b12a
	b999, b13, b1, b2, b12a
	b999, b1, b3, b10
	b999, b13, b1, b3, b10
	b999, b13, b1, b3, for5, b5, b6
	b999, b1, b3, b10, for5, b5, b6
	b999, b13, b1, b3, b10, for5, b5, b6
	b999, b1, b3, for5, b5, b6, b5, b6
	b999, b13, b1, b3, for5, b5, b6, b5, b6
	b999, b1, b3, b10, for5, b5, b6, b5, b6
	b999, b13, b1, b3, b10, for5, b5, b6, b5, b6
$du(b11, d999)$	prázdná
$du(b12b, d999)$	prázdná
$du(b9, d999)$	b9, b3, b10
	b9, b3, b10, for5, b5, b6
	b9, b3, b10, for5, b5, b6, b5, b6
	b9, b3, for5, b5, b6
	b9, b3, for5, b5, b6, b5, b6

Cesty $du(n_i, n_j, d999)$

Jedná se o cesty vzhledem k proměnné $d999$, které začínají v uzlu n_i a končí v uzlu n_j .

Označení cesty	Cesta
$du(b999, b12a, d999)$	b999, b1, b2, b12a
	b999, b13, b1, b2, b12a
$du(b999, b10, d999)$	b999, b1, b3, b10
	b999, b13, b1, b3, b10
$du(b999, b6, d999)$	b999, b1, b3, for5, b5, b6
	b999, b13, b1, b3, for5, b5, b6
	b999, b1, b3, b10, for5, b5, b6
	b999, b13, b1, b3, b10, for5, b5, b6
	b999, b1, b3, for5, b5, b6, b5, b6
	b999, b13, b1, b3, for5, b5, b6, b5, b6
	b999, b1, b3, b10, for5, b5, b6, b5, b6
	b999, b13, b1, b3, b10, for5, b5, b6, b5, b6
$du(b9, b10, d999)$	b9, b3, b10
$du(b9, b6, d999)$	b9, b3, b10, for5, b5, b6
	b9, b3, b10, for5, b5, b6, b5, b6
	b9, b3, for5, b5, b6,
	b9, b3, for5, b5, b6, b5, b6

Cesty du(b999, nj, d999)

Cesty od počátečního uzlu *b999* do uzlu *n_j*.

Označení cesty	Cesta
du(b999, b12a, d999)	b999, b1, b2, b12a
	b999, b13, b1, b2, b12a
du(b999, b10, d999)	b999, b1, b3, b10
	b999, b13, b1, b3, b10
du(b999, b6, d999)	b999, b1, b3, for5, b5, b6
	b999, b13, b1, b3, for5, b5, b6
	b999, b1, b3, b10, for5, b5, b6
	b999, b13, b1, b3, b10, for5, b5, b6
	b999, b1, b3, for5, b5, b6, b5, b6
	b999, b13, b1, b3, for5, b5, b6, b5, b6
	b999, b1, b3, b10, for5, b5, b6, b5, b6
	b999, b13, b1, b3, b10, for5, b5, b6, b5, b6
du(b9, b10, d999)	b999, b1, b2, b11, for4, b4, b9, b3, b10
du(b9, b6, d999)	b999, b1, b2, b11, for4, b4, b9, b3, b10, for5, b5, b6
	b999, b1, b2, b11, for4, b4, b9, b3, b10, for5, b5, b6, b5, b6
	b999, b1, b2, b11, for4, b4, b9, b3, for5, b5, b6
	b999, b1, b2, b11, for4, b4, b9, b3, for5, b5, b6, b5, b6

Závěr

Pomocí datového toku pro proměnnou *d999* jsme zjistili, které cesty obsahují její definici a použití. Cesty nesmí procházet přes jinou definici *d999*. Průchodem těchto cest můžeme otestovat chování (datový tok) pro tuto proměnnou.