ZÁKLADY MATEMATICKEJ TEÓRIE PROGRAMOV

Časť 2.1: Správnosť programov – Floydova metóda

Igor Prívara
Inštitút informatiky a štatistiky

FMFI UK Bratislava, Katedra informatiky

Základy teórie programovania

Programové schémy – abstrakcia programov

- základné pojmy, vlastnosti programových schém
- rozhodnuteľnosť vlastností programových schém
- porovnávanie tried programových schém

Správnosť programov – vzhľadom na špecifikácie

- **Floydova metóda** a Hoareova metóda
- indukčné techniky použité pri dokazovaní správnosti
- (systematická) konštrukcia správnych programov
- dokazovanie správnosti rekurzívnych programov

Sémantika programov – formálny význam programov

- princípy operačnej, denotačnej a axiomatickej sémantiky
- algebraická štruktúra sémantických domén
- formálna definícia denotačného a operačného významu imperatívnych a rekurzívnych programov
- porovnanie denotačného a operačného významu programov

Typy a sémantika – využitie typov pri definícii sémantiky

Správnosť programov

Špecifikácia programu – presný popis **čo** má program urobiť:

vstupná podmienka – vymedzuje množinu zmysluplných vstupov,

výstupná podmienka – charakterizuje vlastnosti požadovaných výstupov.

Špecifikačný jazyk – predikátový počet 1. rádu.

Vlastnosti správnych programov – relatívne vzhľadom na danú špecifikáciu, t.j. vstupnú podmienku p a výstupnú podmienku q:

totálna správnosť – vzhľadom na podmienky p, q:
pre každý vstup, spĺňajúci vstupnú podmienku p, sa program P zastaví a výstupné hodnoty spĺňajú výstupnú podmienku q,

zastavenie – vzhľadom na podmienku p:
pre každý vstup, vyhovujúci vstupnej podmienke p sa program P zastaví,

čiastočná správnosť – vzhľadom na podmienky p, q:
pre každý vstup, spĺňajúci vstupnú podmienku p, pre ktorý sa program P zastaví, zodpovedajúce výstupné hodnoty spĺňajú výstupnú podmienku q.

Metódy dokazovania – (čiastočnej) správnosti:

- redukcia na dôkaz platnosti formúl špecifikačného jazyka,
- špecializovaný logický systém pre dokazovanie tzv. invariantných formúl $\{p\}\, P\, \{q\}$.

Floydova metóda

redukcia dôkazu čiastočnej správnosti programu na dokazovanie predikátových formúl:

- program rozdelíme **deliacimi bodmi** na konečné cesty (počiatočný, koncový, resp. vnútorné body),
- pre každý vnútorný deliaci bod sformulujeme **invariant** $I_A(\overline{x}, \overline{y})$ – podmienku, ktorá platí pri každom prechode deliacim bodom; počiatočnému deliacemu bodu zodpovedá vstupná podmienka, koncovému výstupná podmienka,
- ku každej konečnej ceste AB odvodíme a dokážeme **verifikačnú podmienku**: ak pri prechode deliacim bodom A je splnená podmienka I_A potom po prechode cestou AB bude v deliacom bode B splnená podmienka I_B

  $$\forall \overline{x}, \overline{y} \;\; I_A(\overline{x}, \overline{y}) \;\&\; R_{AB}(\overline{x}, \overline{y}) \Rightarrow I_B(\overline{x}, r_{AB}(\overline{x}, \overline{y})).$$

  Sémantické vlastnosti cesty:

  – $R_{AB}(\overline{x}, \overline{y})$ – podmienka pre prechod cesty AB,

  – $r_{AB}(\overline{x}, \overline{y})$ – modifikácia pracovných premenných pri prechode AB.

Veta: Ak pre všetky cesty v programe P (definované deliacimi bodmi) platia zodpovedajúce verifikačné podmienky, potom je program P čiastočne správny.

Dôkaz: tvrdenie vyplýva z tranzitivity implikácie (indukcia vzhľadom na výpočet).

Konštrukcia R_{AB} a r_{AB}

Spätná substitúcia – $R_{AB}^{pred}(\overline{x},\overline{y}) =??$ $r_{AB}^{pred}(\overline{x},\overline{y}) =??$

cesta programom (príkaz)

$R_{AB}^{po}(\overline{x},\overline{y}) = true$ $r_{AB}^{po}(\overline{x},\overline{y}) = \overline{y}$

Princíp konštrukcie – indukcia vzhľadom na spätný prechod cestou: ak po príkaze $R_{AB}(\overline{x},\overline{y})$ a $r_{AB}(\overline{x},\overline{y})$, potom pred ním

- prázdny príkaz: (príkaz skoku)

  $R_{AB}(\overline{x},\overline{y})$ $r_{AB}(\overline{x},\overline{y})$

- priradenie $\overline{y} := \overline{t}(\overline{x},\overline{y})$:

  $R_{AB}(\overline{x},\overline{t}(\overline{x},\overline{y}))$ $r_{AB}(\overline{x},\overline{t}(\overline{x},\overline{y}))$

- T–vetva podmienky $p(\overline{x},\overline{y})$:

  $R_{AB}(\overline{x},\overline{y})\ \&\ p(\overline{x},\overline{y})$ $r_{AB}(\overline{x},\overline{y})$

- F–vetva podmienky $p(\overline{x},\overline{y})$:

  $R_{AB}(\overline{x},\overline{y})\ \&\ \neg p(\overline{x},\overline{y})$ $r_{AB}(\overline{x},\overline{y})$

Verifikačné podmienky programu (1)

Program – počíta $\lfloor\sqrt{x}\rfloor$

P: **begin** $[y_1, y_2, y_3] := [0, 0, 1]$
1: $[y_2] := [y_2 + y_3]$
2: **if** $y_2 > x$ **then goto end**
3: $[y_1, y_3] := [y_1 + 1, y_3 + 2]$
4: **goto** 1
end $[z] := [y_1]$

Cesta B12

B:	**R:** true	**r:** $[0, 1, 1]$
1:	**R:** true	**r:** $[y_1, y_2 + y_3, y_3]$
2:	**R:** true	**r:** $[y_1, y_2, y_3]$

Cesta 23412

2:	**R:** $y_2 \leq x$	**r:** $[y_1 + 1, y_2 + y_3 + 2, y_3 + 2]$
3:	**R:** true	**r:** $[y_1 + 1, y_2 + y_3 + 2, y_3 + 2]$
4:	**R:** true	**r:** $[y_1, y_2 + y_3, y_3]$
1:	**R:** true	**r:** $[y_1, y_2 + y_3, y_3]$
2:	**R:** true	**r:** $[y_1, y_2, y_3]$

Cesta 2E

2:	**R:** $y_2 > x$	**r:** $[y_1]$
E:	**R:** true	**r:** $[y_1]$

Verifikačné podmienky programu (2)

Program – počíta $\lfloor\sqrt{x}\rfloor$

P: **begin** $[y_1, y_2, y_3] := [0, 0, 1]$
1: $[y_2] := [y_2 + y_3]$
2: **if** $y_2 > x$ **then goto end**
3: $[y_1, y_3] :=[y_1 + 1, y_3 + 2]$
4: **goto** 1
end $[z] := [y_1]$

Invarianty

$I_B(x)$: $x \geq 0$
$I_2(x, y_1, y_2, y_3)$: $(y_1^2 \leq x \ \& \ y_2 = (y_1 + 1)^2 \ \& \ y_3 = 2y_1 + 1)$
$I_E(x, z)$: $z^2 \leq x < (z + 1)^2$

Verifikačné podmienky

- cesta B2:

$$\forall x \ [I_B(x) \ \& \ true \Rightarrow I_2(x, 0, 1, 1)]$$

- cesta 22:

$$\forall x, y_1, y_2, y_3 \ [I_2(x, y_1, y_2, y_3) \ \& \ y_2 \leq x \Rightarrow I_2(x, y_1{+}1, y_2{+}y_3{+}2, y_3{+}2)]$$

- cesta 2E:

$$\forall x, y_1, y_2, y_3 \ [I_2(x, y_1, y_2, y_3) \ \& \ y_2 > x \Rightarrow I_E(x, y_1)]$$

Verifikačné podmienky schémy (1)

S: **begin** $[y_1, y_2] := [x, a]$

1: $[y_1] := [g(y_1, y_2)]$
2: **if** $p_1(y_1)$ **then goto** 5
3: $[y_1, y_2] := [f_1(y_1), f_2(y_2)]$
4: **goto** 1
5: $[y_2] := [g(y_2, y_1)]$
6: **if** $p_2(y_2)$ **then goto end**
7: $[y_1, y_2] := [g_1(y_1), g_2(y_2)]$
8: **goto** 1

end $[z] := [g_1(y_1)]$

2:	**R:** $\neg p_2(g(y_2, y_1)) \& p_1(y_1)$	**r:** $[g(g_1(y_1), g_2(g(y_2, y_1))), g_2(g(y_2, y_1))]$
5:	**R:** $\neg p_2(g(y_2, y_1))$	**r:** $[g(g_1(y_1), g_2(g(y_2, y_1))), g_2(g(y_2, y_1))]$
6:	**R:** $\neg p_2(y_2)$	**r:** $[g(g_1(y_1), g_2(y_2)), g_2(y_2)]$
7:	**R:** true	**r:** $[g(g_1(y_1), g_2(y_2)), g_2(y_2)]$
8:	**R:** true	**r:** $[g(y_1, y_2), y_2]$
1:	**R:** true	**r:** $[g(y_1, y_2), y_2]$
2:	**R:** true	**r:** $[y_1, y_2]$

Verifikačné podmienky schémy (2)

S: **begin** $[y_1, y_2] := [x, a]$

1: $[y_1] := [g(y_1, y_2)]$
2: **if** $p_1(y_1)$ **then goto** 5
3: $[y_1, y_2] := [f_1(y_1), f_2(y_2)]$
4: **goto** 1
5: $[y_2] := [g(y_2, y_1)]$
6: **if** $p_2(y_2)$ **then goto end**
7: $[y_1, y_2] := [g_1(y_1), g_2(y_2)]$
8: **goto** 1

end $[z] := [g_1(y_1)]$

Verifikačné podmienky

- cesta B12:

$$\forall x \; [I_B(x) \;\&\; true \Rightarrow I_2(x, g(x, a), a)]$$

- cesta 23412:

$$\forall x, y_1, y_2 \quad [I_2(x, y_1, y_2) \;\&\; \neg p_1(y_1) \Rightarrow I_2(x, g(f_1(y_1), f_2(y_2)), f(y_2))]$$

- cesta 2567812:

$$\forall x, y_1, y_2 \quad [I_2(x, y_1, y_2) \;\&\; \neg p_2(g(y_2, y_1)) \;\&\; p_1(y_1) \Rightarrow I_2(x, g(g_1(y_1), g_2(g(y_2, y_1))), g_2(g(y_2, y_1)))]$$

- cesta 256E:

$$\forall x, y_1, y_2 \; [I_2(x, y_1, y_2) \;\&\; p_2(g(y_2, y_1)) \& \; p_1(y_1) \Rightarrow I_E(x, g_1(y_1))]$$

Dôkaz zastavenia programu

Dobre "založená" množina – množina W s čiastočným usporiadaním $\succ$, v ktorom neexistuje nekonečná klesajúca postupnosť $x_1 \succ x_2 \succ \cdots \succ x_n \succ \cdots$ prvkov z W

Zobrazenie stavu výpočtu do dobre založenej množiny W –

- ku každému deliacemu bodu A priradíme funkciu

$$u_A : D_{\overline{x}} \times D_{\overline{y}} \mapsto W$$

- ku každej konečnej ceste AB postavíme verifikačnú podmienku konvergencie

$$\forall \overline{x}, \overline{y} \;\; p(\overline{x}) \;\&\; R_{AB}(\overline{x}, \overline{y}) \Rightarrow [u_A(\overline{x}, \overline{y}) \succ u_B(\overline{x}, r_{AB}(\overline{x}, \overline{y}))]$$

Veta: Ak sú všetky podmienky ukončenia, skonštruované k programu P a vstupnej podmienke p, pravdivé potom sa program P zastaví pre všetky vstupy spĺňajúce podmienku p

Dôkaz: tvrdenie vyplýva z tranzitivity implikácie (indukcia vzhľadom na cesty v programe)

Poznámka: zoslabenie verifikačných podmienok ukončenia pomocou "špeciálnych" invariantov $CI_A(\overline{x}, \overline{y})$

$$\forall \overline{x}, \overline{y} \;\; [CI_A(\overline{x}, \overline{y}) \Rightarrow u_A(\overline{x}, \overline{y}) \in W]$$

$$\forall \overline{x}, \overline{y} \;\; [CI_A(\overline{x}, \overline{y}) \;\&\; R_{AB}(\overline{x}, \overline{y})] \Rightarrow [u_A(\overline{x}, \overline{y}) \succ u_B(\overline{x}, r_{AB}(\overline{x}, \overline{y}))]$$

Príklad dôkazu zastavenia programu:

Program – počíta $\lfloor\sqrt{x}\rfloor$ pre $x \geq 0$

P: **begin** $[y_1, y_2, y_3] := [0, 0, 1]$
1: $[y_2] := [y_2 + y_3]$
2: **if** $y_2 > x$ **then goto end**
3: $[y_1, y_3] := [y_1 + 1, y_3 + 2]$
4: **goto** 1
end $[z] := [y_1]$

Invarianty

$CI_B(x)$: $x \geq 0$
$CI_1(x, y_1, y_2, y_3)$: $(y_2 \leq x \ \& \ y_3 > 0)$

Zobrazenie

$u_1(x, y_1, y_2, y_3)$: $x - y_2$

Verifikačné podmienky ukončenia

- $\forall x \ [CI_B(x) \ \& \ true \Rightarrow CI_1(x, 0, 0, 1)]$
- $\forall x, \overline{y} \ [CI_1(x, \overline{y}) \ \& \ y_2 + y_3 \leq x \Rightarrow CI_1(x, y_1 + 1, y_2 + y_3, y_3 + 2)]$
- $\forall x, \overline{y} \ [CI_1(x, \overline{y}) \ \Rightarrow \ u_1(x, \overline{y}) \in N]$
- $\forall x, \overline{y} \ [CI_1(x, \overline{y}) \ \& \ y_2 + y_3 \leq x] \Rightarrow$
  $[u_1(x, \overline{y}) \succ u_1(x, y_1 + 1, y_2 + y_3, y_3 + 2)]$