

Správnost programů

Z FAV wiki

Testováním programu, tj. ověřením, že pro daný vstup získáme správný výsledek, nemůžeme obecně prokázat správnost programu. Množství vstupů je většinou tak velké, že testování není možné ani za použití velice rychlých počítačů. Na druhé straně testování může alespoň prokázat chybu v programu. Protože empirickým testováním nemůžeme zaručit správnost programu, musíme ho nahradit analytickým přístupem pro porozumění správnosti programu. Sekvence příkazů (přřazení, alternativy a cyklu) umožňuje dokázání správnosti programu tak, že můžeme napsat tvrzení o hodnotách proměnných před vykonáním příkazu, které nazýváme předpoklad a tvrzení po jeho vykonání, které nazýváme důsledek.

en → cs
Sebou

Příkazy sekvenčně

Například pokud máme dva příkazy sekvenčně za sebou, tak vyjdeme z požadovaného důsledku druhého příkazu. Na základě důsledku stanovíme požadovaný předpoklad pro druhý příkaz, což je také požadovaný důsledek pro první příkaz atd. až získáme hodnoty možných vstupů.

Příkazy alternativy

Příkazy alternativy zachovávají sekvenční vykonávání příkazu.

Příkazy cyklu

Pro pochopení příkazů opakování zavedeme pojem invariant cyklu. Pro invariant cyklu musíme obecně ukázat tři vlastnosti:

- **Inicializace** - je splněn před vykonáním prvního cyklu.
- **Udržování** - je-li splněn pře vykonáním cyklu, zůstává splněn i po něm.
- **Skončení** - když příkaz cyklu skončí, invariant nám ukáže správnost algoritmu (dokážeme-li, že inicializace platí, a cyklus je udržován pro všechny iterace).

Příklad na řazení pole vkládáním:

- **Inicializace**: Po začátečním přiřazení $i = 1$ je pole s prvky s indexy $0, \dots, i-1$ osahující jeden prvek triviálně seřazeno.
- **Udržování**: Cyklus while posouvá již seřazené prvky $a[i-1], a[i-2], \dots$ o jednu pozici doprava dokud se nenajde správné místo pro $a[i]$. Výsledkem je, že prvky pole $a[0] \dots a[i]$ jsou seřazené. Invariant je tedy splněn i pro následující opakování, kdy i je v hlavičce cyklu inkrementováno.
- **Skončení**: Cyklus for skončí když $i = n$, kde n je počet prvků pole. Dosazením do textu invariantu cyklu dostáváme, že prvky s indexy $0 \dots n-1$ obsahují seřazené původní prvky pole, co je to co jsme požadovali.