Dowodzenie poprawności programów

Wstęp do Informatyki i Programowania

Maciek Gębala

7 listopada 2024

Notatki

Niezmiennik i skończoność pętli

Poprawność programu

Na wykładzie tym zajmiemy się dwoma zagadnieniami:

- czy pętla jest poprawna?
- czy pętla zawsze kończy pracę?

Definicja: Częściowa poprawność

Mówimy, że program S jest częściowo poprawny gdy dla danych spełniających warunek P, jeśli program S kończy pracę, to jego wyniki spełniają warunek Q.
Jeśli program nie kończy pracy, to jest częściowo poprawny.

Definicja: Pełna poprawność

Mówimy, że program S jest w pełni poprawny gdy, jeśli dane spełniają warunek P, to program S kończy pracę i jego wyniki spełniają warunek Q.
Jeśli program nie kończy pracy, to nie jest w pełni poprawny.

Notatki

Niezmiennik i skończoność pętli

Dla pętli istnieje potencjalnie nieskończenie wiele ścieżek obliczeń.

Jak udowodnić poprawność nieskończenie wielu ścieżek obliczeń?

Dowody przez indukcję.

Rozpatrzmy następującą pętlę

```
1: while W do
2:     S
3: end while
```

Gdzie W jest warunkiem sterującym pętli a S ciągiem instrukcji.

Notatki

Niezmiennik i skończoność pętli

Zawsze istnieje warunek P taki, że:

```
1: – { zakładamy ( P ∧ W ) }
2: S
3: – { spełnione jest ( P ) }
```

to znaczy, jeśli spełniony jest warunek P oraz W, to po wykonaniu ciągu instrukcji S spełniony jest warunek P (takim warunkiem jest np. True).

Można udowodnić, że taki warunek P ma również taką własność:

```
1: – { spełnione jest ( P ) }
2: while W do
3:     S
4: end while
5: – { spełnione jest ( P ∧ ¬W ) }
```

Warunek P o powyższej własności nazywa się *niezmiennikiem pętli*.

Notatki

Niezmiennik i skończoność pętli

Nazwa *niezmiennik* podkreśla, że warunek ten jest zachowany podczas wykonywania pętli (nie zmienia się).

Przyjmijmy, że w pętli zmieniają swoje wartości zmienne $x_1, x_2, \ldots, x_n$

Warunek P będziemy zapisywać jako $P(x_1, x_2, \ldots, x_n)$.

Notatki

Niezmiennik i skończoność pętli

Aby udowodnić, że $P(x_1, x_2, \ldots, x_n)$ jest niezmiennikiem będziemy stosować indukcję :

- w pierwszym kroku będziemy sprawdzać, czy po dojściu do pętli (a przed rozpoczęciem jej wykonywania) dane spełniają warunek $P(x_1, x_2, \ldots, x_n)$ (w tym celu podstawimy za zmienne x_1, x_2, ..., x_n wartości jakie mają przy dojściu do pętli i sprawdzimy czy zachodzi warunek P).
- w drugim kroku będziemy udowadniać następującą implikację:

$$P(x_1, x_2, \ldots, x_n) \wedge W(x_1, x_2, \ldots, x_n) \wedge Q(x_1, x_2, \ldots, x_n, x_1', x_2', \ldots, x_n') \implies P(x_1', x_2', \ldots, x_n')$$

gdzie warunek $Q(x_1, x_2, \ldots, x_n, x_1', x_2', \ldots, x_n')$ wyraża zależność nowych wartości zmiennych (te z primami) od starych wartości (te bez primów)

Notatki

Niezmiennik i skończoność pętli

Przykład

Dla poniższej pętli

1: **while** W **do**
2: $X \leftarrow X + 1$
3: $Y \leftarrow Y + X$
4: **end while**

warunek $Q(X, X', Y, Y')$ jest następujący $(X' = X + 1 \wedge Y' = Y + X')$ (zwróć uwagę, że do starej wartości Y dodawana jest nowa wartość X' obliczona w poprzedniej instrukcji).

Jeśli powiedzie się dowód indukcyjny, to udowodniliśmy, że po wykonaniu pętli **while** nadal zachodzi warunek P, czyli jest on niezmiennikiem pętli.

Notatki

Niezmiennik i skończoność pętli

Kluczową rolę odgrywa to, jaki warunek P będziemy dowodzić.

Jak wcześniej wspomniano taki warunek zawsze istnieje ale warunek True jest bezużyteczny gdyż wszystkie wartości danych go spełniają, więc nic nie mówi on o działaniu pętli.

Najlepiej byłoby za P przyjąć warunek spełniony przez jak najmniejszą liczbę danych (jak najsilniejszy), dlatego przez niezmiennik pętli będziemy uważać najsilniejszy warunek spośród tych, które są zachowane podczas działania pętli.

Notatki

Niezmiennik i skończoność pętli

Skąd wziąć taki najsilniejszy warunek?

Pisząc pętlę, myślimy jak ona ma działać by uzyskać pożądany efekt i prawie zawsze wyobrażamy sobie taki niezmiennik.

Dla pętli napisanych przez innych sytuacja jest bardziej skomplikowana: najlepiej prześledzić działanie pętli analizując ślad wykonania jej pierwszych kilku/kilkunastu iteracji i obserwując jak zmieniały się dane w kolejnych iteracjach można starać się odgadnąć postać niezmiennika.

Po sformułowaniu warunku P należy jeszcze udowodnić indukcyjnie, że faktycznie jest to niezmiennik.

Notatki

Niezmiennik i skończoność pętli - przykład

Przykład: iloraz całkowity

W poniższym fragmencie kodu dane są wartości zmiennych całkowitoliczbowych $x \geqslant 0$ i $y > 0$.

```
1: q ← 0
2: r ← x
3: while r ⩾ y do
4:     r ← r − y
5:     q ← q + 1
6: end while
```

Udowodnimy, że powyższa pętla wylicza iloraz całkowity q i resztę r z dzielenia x przez y.

Pokażemy indukcyjnie, że niezmiennikiem jest

$$P(q, r) \equiv (x = q \cdot y + r) \wedge (0 \leqslant r)$$

Notatki

Niezmiennik i skończoność pętli - przykład

W pierwszym kroku sprawdzimy czy wartości $q = 0$ i $r = x$ spełniają warunek niezmiennika:

$$P(0, x) \equiv (x = 0 \cdot y + x) \wedge (0 \leqslant x) \equiv (x = x) \wedge (0 \leqslant x) \equiv (0 \leqslant x)$$

W drugim kroku pokażemy prawdziwość implikacji

$$P(q, r) \wedge (r \geqslant y) \wedge Q(q, q', r, r') \implies P(q', r'),$$

gdzie

$$Q(q, q', r, r') \equiv (r' = r - y) \wedge (q' = q + 1)$$

Notatki

Niezmiennik i skończoność pętli - przykład

Założenia

$$(x = q \cdot y + r) \wedge (0 \leqslant r) \wedge (r \geqslant y) \wedge (r' = r - y) \wedge (q' = q + 1)$$

Teza

$$(x = q' \cdot y + r') \wedge (0 \leqslant r')$$

Dowód

$$x = q \cdot y + r = (q' - 1) \cdot y + r = (q' - 1) \cdot y + (r' + y) = q' \cdot y + r'$$

$$r' = r - y \geqslant 0$$

zatem nowe wartości q' i r' spełniają warunek P

Notatki

Niezmiennik i skończoność pętli - przykład

Dowód

Po wyjściu z pętli zachodzi

$$P(q,r) \wedge (r < y) \equiv (x = q \cdot y + r) \wedge (0 \leqslant r < y)$$

Skończoność wykonania pętli wynika z kolejnego zmniejszania się r i dojścia do kresu dolnego mniejszego niż y.

Notatki

Niezmiennik i skończoność pętli - przykład

Przykład: potęgowanie całkowite

W poniższym fragmencie kodu dane są wartości zmiennych $x > 0$ i $n \geqslant 0$

```
1:  y ← 1
2:  p ← x
3:  i ← n
4:  while i ≠ 0 do
5:    if i mod 2 = 0 then
6:      p ← p · p
7:      i ← i/2
8:    else
9:      y ← y · p
10:     i ← i − 1
11:   end if
12: end while
```

Udowodnimy, że powyższa pętla wylicza x^n.

Pokażemy indukcyjnie, że niezmiennikiem jest $P(y,p,i) \equiv y \cdot p^i = x^n$.

Notatki

Niezmiennik i skończoność pętli - przykład

W pierwszym kroku sprawdzamy czy wartości $y = 1$, $p = x$ i $i = n$ spełniają warunek niezmiennika: $P(1,x,n) \equiv 1 \cdot x^n = x^n \equiv x^n = x^n$

W drugim kroku pokazujemy prawdziwość implikacji

$$P(y,p,i) \wedge (i \neq 0) \wedge Q(y,y',p,p',i,i') \implies P(y',p',i'),$$

gdzie $Q(y,y',p,p',i,i')$ opisuje jak w jednym przebiegu pętli zmieniają się wartości zmiennych y, p oraz i.

Notatki

Niezmiennik i skończoność pętli - przykład

Zmiany zmiennych zależą od parzystości wartości zmiennej i, zatem dowód będzie składał się z dwóch przypadków:

- i jest parzyste
  - założenia: $(y \cdot p^i = x^n) \wedge (i \neq 0) \wedge (y' = y) \wedge (p' = p^2) \wedge (i' = i/2)$
  - teza: $y' \cdot p'^{i'} = x^n$
  - dowód: $y' \cdot p'^{i'} = y \cdot (p^2)^{i/2} = y \cdot p^i = x^n$
- i jest nieparzyste
  - założenia: $(y \cdot p^i = x^n) \wedge (i \neq 0) \wedge (y' = y \cdot p) \wedge (p' = p) \wedge (i' = i - 1)$
  - teza: $y' \cdot p'^{i'} = x^n$
  - dowód: $y' \cdot p'^{i'} = (y \cdot p) \cdot p^{i-1} = y \cdot p^i = x^n$

Po wyjściu z pętli zachodzi warunek $P(y,p,i) \wedge (i = 0)$, zatem $(y \cdot p^i = x^n) \wedge (i = 0) \implies y = x^n$

Notatki

Skończoność pętli

Samo udowodnienie niezmiennika nie gwarantuje, że program jest w pełni poprawny. Należy jeszcze udowodnić, że dla wszystkich poprawnych danych pętla kończy pracę.

Aby pokazać, że pętla kończy pracę należy podać wyrażenie o wartościach całkowitych ograniczonych z dołu, które przy każdej iteracji pętli maleją.

Porządek $<$ na liczbach naturalnych jest dobrze ufundowany, gdyż nie istnieje nieskończony ciąg malejący o wartościach naturalnych.

Analogicznie można dla dowodu podać wyrażenie o wartościach całkowitych ograniczonych z góry, które przy każdej iteracji pętli rosną (wówczas maleje dystans do górnego ograniczenia).

Notatki

Ciekawostka

Hipoteza Collatza (hipoteza $3n+1$)

W roku 1937 niemiecki matematyk Lothar Collatz sformułował hipotezę, że poniższy ciąg zawsze osiąga wartość 1 dla dowolnej początkowej wartości całkowitej $c_0 > 1$

$$c_{n+1} = \begin{cases} c_n/2 & \text{gdy } c_n \text{ jest parzyste} \\ 3 \cdot c_n + 1 & \text{gdy } c_n \text{ jest nieparzyste} \end{cases}$$

Hipoteza pozostaje do dziś nierozstrzygnięta (sprawdzono liczby mniejsze od $5 \cdot 2^{60} \approx 5.764 \cdot 10^{18}$).

Notatki

Notatki

Notatki