PÁRHUZAMUS PROGRAMOK HELYESSÉGE

Bevezetés

- Többféle változatot vizsgálunk
  - Először megköveteljük a párhuzamosan futó folyamatoktól a diszjunktságot.
  - Utána a közös változót használó párhuzamos folyamatokkal foglalkozunk.
- Elemzés
  - Helyesség
  - Interferencia
  - Kritikus szakaszok kölcsönös kizárása
  - Holtpontmentesség

Helyességi kategóriák

- A $\{Q\}\ S\ \{R\}$ parciálisan helyességi tétel azt fejezi ki, hogy az S program Q feltételt kielégítő állapotból elindulva – amennyiben az S program terminál – egy R feltételt kielégítő állapotban áll meg.
- A $\{\{Q\}\}\ S\ \{\{R\}\}$ gyenge teljes helyességi tétel azt fejezi ki, hogy az S program Q feltételt kielégítő állapotból elindulva – amennyiben az S program nem kerül holtpont helyzetbe – egy R feltételt kielégítő állapotban fog megállni (nem esik végtelen ciklusba, nem lesz hibás a működés).
- A $\{\{Q\}\}\ S\ \{\{R\}\}$ teljes helyességi tétel azt fejezi ki, hogy az S program Q feltételt kielégítő állapotból elindulva egy R feltételt kielégítő állapotban fog megállni. (gyenge teljes helyesség + holtpontmentesség)

Annotáció

- Jelöljük S^*-gal S-nek a $\{Q\}S\{R\}$ bizonyításához szükséges segédállításokkal kiegészített változatát, ún. annotációját.
- Megengedett, hogy az annotált programot olyan értékadásokkal egészítsük ki, amelyek újonnan felvett segédváltozóknak adnak értéket. Ezeknek a segédváltozóknak az a szerepe, hogy megjelenjenek a $\{Q\}S^*\{R\}$ tétel segédállításaiban.
- Nyilvánvaló, hogy ha egy annotált programra belátható a $\{Q\}S^*\{R\}$ tétel, akkor az eredeti programra teljesül a $\{Q\}S\{R\}$.

Diszjunkt programok fogalma

- Jelölje az S program esetén
  - $var(S)$ ~ az S összes változóját
  - $change(S)$ ~ az S-beli értékadások baloldalán szereplő változókat
- Az S_1 és S_2 programok diszjunktak, ha $var(S_1) \cap change(S_2)=\emptyset$ és $var(S_2) \cap change(S_1)=\emptyset$
- Az E kifejezés és S program diszjunktak, ha $free(E) \cap change(S)=\emptyset$
  - Az E egy elsőrendű logikai kifejezés, $free(E)$ ennek szabad (nem kvantált) változói.

Diszjunkt párhuzamossági szabály

$\{Q_i\}S_i\{R_i\}$ $i=1..n$ és
$\forall i,j\in\{1...n\}, i\neq j$ -re a Q_i is, R_i is diszjunkt S_j-vel

$\{Q_1 \wedge ... \wedge Q_n\}$ **parbegin** $S_1 \parallel \ldots \parallel S_n$ **parend** $\{R_1 \wedge ... \wedge R_n\}$

- Bizonyítás:
  - $\forall i,j\in[1..n]$,$i\neq j$ -re $\exists S_i^*$ annotáció, hogy $\forall P\in S_i^*$ állításra és $\forall u\in S_j$ utasítására : $free(P)\cap change(u)=\emptyset$
  - $\forall i\in[1..n]$-re: $\{P_1 \wedge ... \wedge Q_i \wedge ... \wedge P_n\}\ S_i\ \{P_1 \wedge ... \wedge R_i \wedge ... \wedge P_n\}$, ahol $\forall j\in[1..n]\setminus\{i\}$ -re $P_j\in S_j^*$ állítása
  - Amennyiben a párhuzamos blokk $Q_1 \wedge ... \wedge Q_n$-ből indulva befejeződik, akkor az összes S_i $(i=1..n)$ is befejeződik, és ekkor egy $R_1 \wedge ... \wedge R_n$-t kielégítő állapotba jutunk.

Szekvencializációs szabály

$\{Q\}S_1 ; \dots ; S_n \{R\}$ és
$\forall i,j \in \{1 \dots n\}, i \neq j$ -re az S_i és is S_j diszjunkt

$\{Q\}$ **parbegin** $S_1 \parallel \dots \parallel S_n$ **parend** $\{R\}$

- Bizonyítás:
  - $S_1 , \dots , S_n$ közös változói ($\underline{x}$) kizárólag bemenő változók lehetnek a diszjunktság miatt; minden más valamelyik S_i kizárólagos változója ($\underline{v}_i$).
    $Q(\underline{v}) = Q_0(\underline{x}) \wedge Q_1(\underline{x}, \underline{v}_1) \wedge \dots \wedge Q_n(\underline{x}, \underline{v}_n)$ /$\underline{v}_i$=change(S_i)
    $R(\underline{v}) = R_1(\underline{x}, \underline{v}_1) \wedge \dots \wedge R_n(\underline{x}, \underline{v}_n)$ /free(Q_i,R_i) $\cap$ $\underline{v}_j$=Ø
  - A $\{Q\}S_1 ; \dots ; S_n \{R\}$ annotálható úgy, hogy $\forall i \in [1..n]$-re
    $\{Q_0 \wedge R_1 \wedge \dots \wedge R_{i-1} \wedge Q_i \wedge \dots \wedge Q_n\}\ S_i\ \{Q_0 \wedge R_1 \wedge \dots \wedge R_i \wedge Q_{i+1} \wedge \dots \wedge Q_n\}$
    fenn álljon, amiből $\{Q_0 \wedge Q_i\}S_i\{R_i\}$ is következik, azaz $\{Q'_i\}S_i\{R_i\}$
  - Innentől pedig az előző szabály alapján adódik a következmény.

Példa

- Lássuk be, hogy az alábbi program kiszámolja az 1-től n-ig (n≥0) vett természetes számok összegét!
  - Q : n=n'
  - R : n=n' ∧ s = (n(n+1))/2 $\quad s = \sum_{j=1}^{n} j$

{n=n'}

```
i1 := 1; i2 := 2; s1 := 0; s2 := 0;
parbegin
  while i1≤n do s1 := s1+i1  i1 := i1+2; od ;
||
  while i2≤n do s2 := s2+i2;  i2 := i2+2; od ;
parend
s := s1+s2 ;
```

{n=n' ∧ s = 1 + 2 + ... + n }

Szekvencia szabálya

{n=n'}

```
i1 := 1; i2 := 2; s1 := 0; s2 := 0;
```

✓

{n=n' ∧ i1=1 ∧ i2= 2 ∧ s1=0 ∧ s2=0}

```
parbegin
  while i1≤n do s1 := s1+i1;  i1 := i1+2; od ;
||
  while i2≤n do s2 := s2+i2;  i2 := i2+2; od ;
parend
```

{n=n' ∧ ps(n) → (s1=1+3+...+n−1 ∧ s2 =2+4+...+n) ∧
ptl(n) → (s1=1+3+...+n ∧ s2 =2+4+...+ n−1) }

```
s := s1+s2 ;
```

✓

{n=n' ∧ s = 1 + 2 + ... + n }

Diszjunkt párhuzamossági szabály alkalmazása

- free(Q_1)=free(R_1)={n,i1,s1} change(S_2)={i2,s2}
- free(Q_2)=free(R_2)={n,i2,s2} change(S_1)={i1,s1}

Q_1 {n=n' ∧ i1=1 ∧ s1=0}

S_1 **while** i1≤n **do** s1:= s1+i1; i1:= i1+2; **od** ;

R_1 {n=n' ∧ ps(n) → s1=1+3+...+n−1 ∧
ptl(n) → s1=1+3+...+n}

Q_2 {n=n' ∧ i2= 2 ∧ s2=0}

S_2 **while** i2≤n **do** s2 := s2+i2; i2 := i2+2; **od** ;

R_2 {n=n' ∧ ps(n) → s2=2+4+...+n ∧
ptl(n) → s2=2+4+...+n−1 }

Ciklus következtetési szabálya

$I_1(n, s1, i1) =$
$n=n' \wedge 1 \le i1 \le n+2 \wedge ptl(i1) \wedge s1=1+3+...+i1-2$

{n=n' ∧ i1=1 ∧ s1=0 } ⇒ I_1(n, s1, i1)

while i1≤n **do** s1:=s1+i1; i1:=i1+2; **od** ;

$I_1(n, s1, i1) \wedge i1 \le n \Rightarrow I_1(n, s1, i1)$

$I_1(n, s1, i1) \wedge i1 > n \Rightarrow$
{n=n' ∧ ps(n)→i1= n+1 ∧ ptl(n)→i1= n+2 ∧ I_1(n, s1, i1)}
azaz
{n=n' ∧ ps(n) → s1 = 1+3+...+n−1 ∧
ptl(n) → s1 = 1+3+...+n }

Ciklus következtetési szabálya

$I_2(n, s2, i2) =$
$n=n' \wedge 1 \le i2 \le n+2 \wedge ps(i2) \wedge s2=2+4+...+i2-2$

{n=n' ∧ i2=2 ∧ s2=0 } ⇒ I_2(n, s2, i2)

while i2≤n **do** s2:=s2+i2; i2:=i2+2; **od** ;

$I_2(n, s2, i2) \wedge i2 \le n \Rightarrow I_2(n, s2, i2)$

$I_2(n, s2, i2) \wedge i2 > n \Rightarrow$
{n=n' ∧ ps(n)→i2= n+2 ∧ ptl(n)→i2= n+1 ∧ I_2(n, s2, i2)}
azaz
{n=n' ∧ ps(n) → s2=2+4+...+n ∧
ptl(n) → s2=2+4+...+n−1 }

Megállás

- $t_1 = n+2-i1$
- $t_2 = n+2-i2$

$I_1(n, s1, i1) \wedge i1 \le n \Rightarrow t_1>0$
$I_2(n, s2, i2) \wedge i2 \le n \Rightarrow t_2>0$

$I_1(n, s1, i1) \wedge i1 \le n \wedge t_1=c \Rightarrow lf(i1:=i1+1, t_1<c)$
$I_2(n, s2, i2) \wedge i2 \le n \wedge t_1=c \Rightarrow lf(i2:=i2+2, t_2<c)$

Közös változók használata és a helyesség

- A közös változók használata esetén egy folyamat működését befolyásolhatja a vele párhuzamosan futó folyamat. Egy folyamat helyességének vizsgálatakor ezt az interferenciát figyelembe kell venni.
- Az interferencia nem kívánt hatását el lehet kerülni, ha
  - bizonyos értékadások szekvenciáját (egyik elront valamit egy párhuzamos folyamatban, de utána a másik ezt helyrehozza, miközben a kettő együtt teljesít egy adott célt) atomi műveletté fűzzük össze.
  - várakoztatunk egy folyamatot addig, amíg a vele párhuzamosan futó folyamat olyan szakaszba nem ér, ahol már nem okozhat neki gondot az interferencia.

Atomi művelet levezetési szabálya

$$\frac{\{\{Q\}\}\ S\ \{\{R\}\}}{\{\{Q\}\}\ \langle S\rangle\ \{\{R\}\}}$$

- Ez a levezetési szabály egyben teljes helyességi szabály is.
- A szabály helyessége az atomi művelet szemantikája alapján igazolható.

Várakozó utasítás levezetési szabálya

$$\frac{\{\{Q \wedge \alpha\}\}\ S\ \{\{R\}\}}{\{\{Q\}\}\ \textbf{await}\ \alpha\ \textbf{then}\ S\ \{\{R\}\}}$$

- Ez a levezetési szabály egyben teljes helyességi szabály is.
- A szabály helyessége az atomi művelet szemantikája alapján igazolható.

Utasítás interferencia-mentessége

- Legyen u egy értékadás vagy értékadást is tartalmazó atomi művelet* a pre_u előfeltétellel.
- Az u nem interferál a $\{Q\}\ S\ \{R\}$ parciális helyességi tétellel, ha van olyan S* annotáció, hogy az S* minden nem atomi műveletbe zárt s utasításának $pre_s \in S^*$ előfeltételére
  $\{pre_u \wedge pre_s\}\ u\ \{pre_s\}$ és
  $\{pre_u \wedge R\}\ u\ \{R\}$
- Az u nem interferál a $\{\{Q\}\}\ S\ \{\{R\}\}$ gyenge teljes helyességi tétellel, ha az előzőek gyenge teljes tételekként is teljesülnek, továbbá
  $\{\{pre_u \wedge t=t_0\}\}\ u\ \{\{t \le t_0\}\}$,
  ahol t a befejeződést igazoló terminális függvény.

* A Q előfeltételű **await** α **then** S **ta** utasítást egy $Q \wedge \alpha$ előfeltételű <S> atomi műveletnek kell tekinteni.

Megjegyzés

- Az interferencia-mentességet mindig a $\{\{Q\}\}\ S\ \{\{R\}\}$ tétel egy konkrét bizonyítására, egy konkrét S* annotációra vizsgáljuk.
- Elképzelhető, hogy az egyik annotációra nem tudjuk kimutatni az interferencia-mentességét, de egy másikra már igen. Ha egy $\{\{pre_u \wedge pre_s\}\}\ u\ \{\{pre_s\}\}$ alakú állítás nem teljesül, mert az u elrontja a pre_s-t, akkor meg kell vizsgálni, hogy van-e olyan másik annotáció, amelyben pre_s helyett olyan pre'_s segédállítás szerepel, amelyre $\{\{pre_u \wedge pre'_s\}\}\ u\ \{\{pre'_s\}\}$.
- Néha segít, ha
  - vagy $pre'_s \equiv pre_s \vee post_u$
  - vagy $pre_u \wedge pre'_s \equiv \downarrow$

Helyességi tételek interferencia-mentessége

- Adottak a $\{Q_k\}\ S_k\ \{R_k\}$ $(k=1...n)$ parciális helyességi tételek az annotációjukkal. Ezek interferencia-mentesek, ha $\forall i,j \in \{1...n\}$, $i \neq j$-re az S_i-nek az összes u_i nem atomi értékadása illetve atomivá tett művelete az adott annotációbeli pre_{u_i} előfeltétel mellett nem interferál a $\{Q_j\}\ S_j\ \{R_j\}$ tétellel.
- Adottak a $\{\{Q_k\}\}\ S_k\ \{\{R_k\}\}$ $(k=1...n)$ gyengén teljes helyességi tételek az annotációjukkal. Ezek interferencia-mentesek, ha $\forall i,j \in \{1...n\}$, $i \neq j$-re az S_i-nek az összes u_i nem atomi értékadása illetve atomivá tett művelete az adott annotációbeli pre_{u_i} előfeltétel mellett nem interferál a $\{\{Q_j\}\}\ S_j\ \{\{R_j\}\}$ tétellel.

Párhuzamos blokk levezetési szabálya

- Parciális illetve gyenge teljes helyesség esetére:

$\{Q_1\}\ S_1\{R_1\}, \dots, \{Q_n\}\ S_n\{R_n\}$ igazak és interferencia-mentesek

---

$\{Q_1 \wedge \dots \wedge Q_n\}$ **parbegin** $S_1 \parallel \dots \parallel S_n$ **parend** $\{R_1 \wedge \dots \wedge R_n\}$

- Bizonyítás:
  Az interferencia-mentesség miatt bárhogyan is fésüljük össze a komponensek utasításait minden $i \in \{1...n\}$-re teljesül, hogy a $Q_1 \wedge \dots \wedge Q_n$-ből indított a párhuzamos program az S_i befejeződésekor egy $P_1 \wedge \dots \wedge R_i \wedge \dots \wedge P_n$-beli állapotban lesz, és a program ezt követően is csak olyan állapotokon keresztül haladhat tovább, ahol R_i teljesül.

Párhuzamos blokk levezetési szabálya

- Teljes helyesség esetére:

$\{\{Q_1\}\}S_1\{\{R_1\}\}, \dots, \{\{Q_n\}\}S_n\{\{R_n\}\}$ igazak és interferencia-mentesek és $S_1, \dots, S_n$-ből képzett párhuzamos blokk holtpontmentes

---

$\{\{Q_1 \wedge \dots \wedge Q_n\}\}$ **parbegin** $S_1 \parallel \dots \parallel S_n$ **parend** $\{\{R_1 \wedge \dots \wedge R_n\}\}$

- Bizonyítás:
  A parciális helyességet már bizonyítottuk, most csak a leállással kel foglakozni.
  A párhuzamos program leállásához egyrészt kell, hogy minden komponens leálljon (ez feltételként adott), másrészt, hogy ne alakuljon ki körkörös várakozás, azaz holtponthelyzet.

Teljes vs. gyenge teljes helyesség

- Egyszerűbb, ha feltesszük, hogy a programunk egyetlen párhuzamos blokkból áll, ilyenkor nincs különbség a komponens programok gyengén teljes és teljes helyessége között, és a komponensek között kell csak vizsgálni a holtpontmentességet.
- Ha beágyazott párhuzamos blokkokat is megengedünk, akkor is elég a komponensekre csak a gyenge teljes helyességet igazolni, majd az egész programra egyszer ellenőrizni a holtpontmentességet.

1. Példa

- Bizonyítsuk be az alábbi parciális helyességi tételt:
  - $\{x = 0\}$ **parbegin** $\langle x := x+2 \rangle \parallel \langle x := x+3 \rangle$ **parend** $\{x = 5\}$
- A program ágak parciálisan helyesek:
  Az értékadás és az atomi művelet szabályai miatt.
  - $\{\ x = 0 \vee x = 3\ \} \langle x := x+2 \rangle \{\ x = 2 \vee x = 5\ \}$
  - $\{\ x = 0 \vee x = 2\ \} \langle x := x+3 \rangle \{\ x = 3 \vee x = 5\ \}$

  továbbá $x=0 \Rightarrow (x=0 \vee x=3) \wedge (x=0 \vee x=2)$
  és $(x=2 \vee x=5) \wedge (x=3 \vee x=5) \Rightarrow x=5$
- Az interferencia-mentességéhez az alábbiakat kell bizonyítani:
  - $\{(x=0 \vee x=3) \wedge (x=0 \vee x=2)\} \langle x := x+2 \rangle \{x=0 \vee x=2\}$
  - $\{(x=0 \vee x=3) \wedge (x=3 \vee x=5)\} \langle x := x+2 \rangle \{x=3 \vee x=5\}$
  - $\{(x=0 \vee x=2) \wedge (x=0 \vee x=3)\} \langle x := x+3 \rangle \{x=0 \vee x=3\}$
  - $\{(x=0 \vee x=2) \wedge (x=2 \vee x=5)\} \langle x := x+3 \rangle \{x=2 \vee x=5\}$

  Az értékadás és az atomi művelet szabályai miatt.

2. Példa

- Bizonyítsuk be az alábbi parciális helyességi tételt:
  - $\{x=0\}$ **parbegin** $x := x+2 \parallel x := 0$ **parend** $\{x=0 \vee x=2\}$
- A program ágak parciálisan helyesek:
  - $\{\ x=0\ \}\ x := x+2\ \{\ x=2\ \}$
  - $\{\ \text{igaz}\ \}\ x := 0\ \{\ x=0\ \}$

  és $x=0 \Rightarrow x=0 \wedge \text{igaz}$ valamint $(x=2 \wedge x=0) \Rightarrow (x=0 \vee x=2)$
  de a tételek nem interferencia-mentesek, mert például :
  - $\{\text{igaz} \wedge x=2\}\ x := 0\ \{x=2\}$ nem áll fenn.
- Válasszunk más részállításokat a bizonyításához (bővítés R-rel)
  - $\{\ x=0\ \}\ x := x+2\ \{\ x=2 \vee x=0\ \}$
  - $\{\ \text{igaz}\ \}\ x := 0\ \{\ x=0 \vee x=2\ \}$

  és ekkor az interferencia-mentesség is belátható.

3. Példa

- Bizonyítsuk be az alábbi parciális helyességi tételt:
  - {x=a ∧ y=b}
    parbegin x := f(x,y) ‖ y:= g(x,y) **parend**
  - {(x=f(a,b) ∧ y=g(f(a,b),b)) ∨ (x=f(a,g(a,b)) ∧ y=g(a,b)) }
- Vezessünk be segédváltozókat a részeredmények jelzésére abból a célból, hogy a részállításokat helyesen fogalmazzuk majd meg:
  - $kész_1$ váltson igaz-ra, ha az első értékadás végrehajtódott
  - $kész_2$ legyen igaz-ra, ha a második értékadás végrehajtódott

```
kész1 := hamis; kész2 := hamis;
parbegin
    x, kész1 := f(x,y), igaz
‖
    y, kész2 := g(x,y), igaz
parend
```


3. Példa: annotálás

```
{ x=a ∧ y=b }
kész1 := hamis; kész2 := hamis;
{ x=a ∧ kész1=↓ ∧ y=b ∧ kész2=↓ }
parbegin
   { kész1=↓ ∧ ¬kész2 → (x=a ∧ y=b) ∧ kész2 → (x=a ∧ y=g(a,b)) }
   x, kész1 := f(x,y), igaz
   { kész1=↑ ∧ ¬kész2 → (x=f(a,b) ∧ y=b) ∧
     kész2 → ((x=f(a,g(a,b)) ∧ y=g(a,b)) ∨ (x=f(a,b) ∧ y=g(f(a,b),b))) }
‖
   { kész2=↓ ∧ ¬kész1 → (x=a ∧ y=b) ∧ kész1 → (x=f(a,b) ∧ y=b) }
   y, kész2 := g(x,y), igaz
   { kész2=↑ ∧ ¬kész1 → (x=a ∧ y=g(a,b)) ∧
     kész1 → ((x=f(a,g(a,b)) ∧ y=g(a,b)) ∨ (x=f(a,b) ∧ y=g(f(a,b),b))) }
parend
{ (x=f(a,b) ∧ y=g(f(a,b),b) ) ∨ (x=f(a,g(a,b)) ∧ y=g(a,b)) }
```


3. Példa: segédváltozó elhagyása

```
{x=a ∧ y=b }                                                          //Q
parbegin
   { x=a ∧ ( y=b ∨ y=g(a,b)) }                                        // q1
   x := f(x,y)
   { (x=f(a,b) ∧y=b)) ∨ ((x=f(a,g(a,b)) ∧ y=g(a,b)) }                // r1
‖
   { y=b ∧ (x=a ∨ x=f(a,b)) }                                         // q2
   y := g(x,y)
   { (x=a ∧ y=g(a,b)) ∨ (x=f(a,b) ∧ y=g(f(a,b),b)) }                  // r2
parend
{(x=f(a,b) ∧ y=g(f(a,b),b) ) ∨ (x=f(a,g(a,b)) ∧ y=g(a,b)) }          //R
```

- A résztételek helyessége könnyen belátható, de az interferencia-mentesség nem (r_1 , r_2).

3. Példa befejezése

- Bővítsük az értékadások utófeltételeit R-rel:

```
{x=a ∧ y=b }                                                          //Q
parbegin
   { x=a ∧ ( y=b ∨ y=g(a,b)) }                                        // q1
   x := f(x,y)
   { (x=f(a,b) ∧y=b)) ∨ (x=f(a,g(a,b)) ∧ y=g(a,b)) ∨
                        (x=f(a,b) ∧ y=g(f(a,b),b) ) }                 // r'1
‖
   { y=b ∧ (x=a ∧ x=f(a,b)) }                                         // q2
   y := g(x,y)
   { (x=a ∧ y=g(a,b)) ∨ (x=f(a,b) ∧ y=g(f(a,b),b)) ∨
                        (x=f(a,g(a,b)) ∧ y=g(a,b)) }                  // r'2
parend
{(x=f(a,b) ∧ y=g(f(a,b),b) ) ∨ (x=f(a,g(a,b)) ∧ y=g(a,b)) }          //R
```

- A résztételek így már interferencia-mentesek is, és
  $\{q_1 \wedge q_2\}\ S\ \{r'_1 \wedge r'_2\}$, továbbá $Q \Rightarrow q_1$, $Q \Rightarrow q_2$, $r'_1 \wedge r'_2 \Rightarrow R$.

4. Példa

- Bizonyítsuk be az alábbi gyenge teljes helyességi tételt (számoljuk ki az x^n hatványt, ahol x valós szám, n természetes szám)!
  - {{ x=x' ∧ n=n' }}

```
z:=1;
parbegin
S1: while n≠ 0 do
        await ptl(n) then n, z:=n−1, z*x ta
    od
‖
S2: while n ≠ 0 do
        await ps(n) then x, n:= x*x, n/2 ta
    od
parend
```

  - {{z= x'^n' }}

4. Példa: programtagok helyessége

```
z:=1;
{{ z*x^n= x'^n'}}
parbegin
```

I≡ {{z * x^n = x'^n'}} , t=n

S1

```
{{ z*x^n= x'^n'}}
while n ≠ 0 do
   {{ I ∧ n ≠ 0}}
   await ptl(n) then
      n, z:=n−1, z*x
   ta
od
{{z= x'^n' }}
```

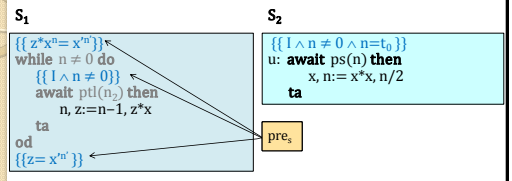
‖

S2

```
{{ z*x^n= x'^n'}}
while n ≠ 0 do
   {{ I ∧ n ≠ 0}}
   await ps(n) then
      x, n:= x*x, n/2
   ta
od
{{z= x'^n' }}
```

```
parend
{{z= x'^n' }}
```

4. Példa: interferencia-mentesség

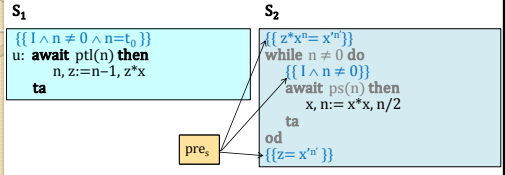


$\{\{ I \wedge n \neq 0 \wedge ps(n) \wedge pre_s \}\} \quad < x, n := x*x, n/2 > \{\{ pre_s \}\}$

$\{\{ I \wedge n \neq 0 \wedge ps(n) \wedge R \}\} \quad < x, n := x*x, n/2 > \{\{ R \}\}$

$\{\{ I \wedge n \neq 0 \wedge ps(n) \wedge n = t_0 \}\} \quad < x, n := x*x, n/2 > \{\{ n \leq t_0 \}\}$

4. Példa: interferencia-mentesség



$\{\{ I \wedge n \neq 0 \wedge ptl(n) \wedge pre_s \}\} \quad < n, z := n-1, z*x > \{\{ pre_s \}\}$

$\{\{ I \wedge n \neq 0 \wedge ptl(n) \wedge R \}\} \quad < n, z := n-1, z*x > \{\{ R \}\}$

$\{\{ I \wedge n \neq 0 \wedge ptl(n) \wedge n = t_0 \}\} \quad < n, z := n-1, z*x > \{\{ n \leq t_0 \}\}$

5. Példa

- Bizonyítsuk be az alábbi gyenge teljes helyességi tételt (a program pozitív egész számot keres az $x:\mathbb{Z}^n$ tömbben)!
  - $\{\{ x=x' \}\}$
  - i,j,ps,pt:ℕ

```
i := 2; j := 1; ps := n+1; pt := n+1;
parbegin
S1: while i<min(ps,pt) do                    i<ps≤pt ∨ i<pt≤ps
        if x[i] > 0 then ps := i else i := i+2 fi
    od
‖
S2: while j<min(ps,pt) do                    j<ps≤pt ∨ j<pt≤ps
        if x[j] > 0 then pt := j else j := j+2 fi
    od
parend
k := min(ps, pt);                  if ps<pt then k := ps else k := pt fi
```

  - $\{\{ x=x' \wedge k \leq n+1 \wedge \forall a \in [1 \,..\, k-1]: x[a] \leq 0 \wedge (k \leq n \rightarrow x[k] > 0) \}\}$

5. Példa: helyesség

$\{\{ x=x' \}\}$
i := 2; j := 1; ps := n+1; pt := n+1;
$\{\{ x=x' \wedge i=1 \wedge j=2 \wedge ps=pt=n+1 \}\}$
parbegin $S_1 \| S_2$ **parend**
$\{\{ x=x' \wedge \max(ps,pt) \leq n+1 \wedge \forall a \in [1 \,..\, \min(ps,pt)-1]: x[a] \leq 0 \wedge (\min(ps,pt) \leq n \rightarrow x[\min(ps,pt)] > 0) \}\}$
k := min(ps, pt);
$\{\{ x=x' \wedge k \leq n+1 \wedge \forall a \in [1 \,..\, k-1]: x[a] \leq 0 \wedge (k \leq n \rightarrow x[k] > 0) \}\}$

5. Példa: programtagok helyessége

$\{\{ x=x' \wedge i=1 \wedge j=2 \wedge ps=pt=n+1 \}\}$
parbegin $S_1 \| S_2$ **parend**
$\{\{ x=x' \wedge \max(ps,pt) \leq n+1 \wedge \forall a \in [1..\min(ps,pt)-1]: x[a] \leq 0 \wedge (\min(ps,pt) \leq n \rightarrow x[\min(ps,pt)] > 0) \}\}$

S_1:
$\{\{ x=x' \wedge i=1 \wedge ps=n+1 \}\}$
while i<min(ps,pt) **do**
  $\{\{ P \wedge t=t_0 \}\}$
  if x[i] > 0 **then**
    $\{\{ P \wedge x[i] > 0 \wedge t=t_0 \}\}$
    ps := i
  else
    $\{\{ P \wedge x[i] \leq 0 \wedge t=t_0 \}\}$
    i := i+2
  fi
od
$\{\{ x=x' \wedge \max(ps,pt) \leq n+1 \wedge \forall a \in [1 \,..\, \min(ps,pt)-1], 2 \nmid a: x[a] \leq 0 \wedge (ps \leq n \rightarrow x[ps] > 0) \}\}$

P ≡ $x=x' \wedge \max(ps,pt) \leq n+1 \wedge i \leq \min(ps,pt) \wedge \forall a \in [1 \,..\, i-1], 2 \nmid a: x[a] \leq 0 \wedge (ps \leq n \rightarrow x[ps] > 0)$
t = n+1−i + ps

5. Példa: programtagok helyessége

$\{\{ x=x' \wedge i=1 \wedge j=2 \wedge ps=pt=n+1 \}\}$
parbegin $S_1 \| S_2$ **parend**
$\{\{ x=x' \wedge \max(ps,pt) \leq n+1 \wedge \forall a \in [1..\min(ps,pt)-1]: x[a] \leq 0 \wedge (\min(ps,pt) \leq n \rightarrow x[\min(ps,pt)] > 0) \}\}$

S_2:
$\{\{ x=x' \wedge j=2 \wedge pt=n+1 \}\}$
while j<min(ps,pt) **do**
  $\{\{ P \wedge t=t_0 \}\}$
  if x[j] > 0 **then**
    $\{\{ P \wedge x[j] > 0 \wedge t=t_0 \}\}$
    ps := j
  else
    $\{\{ P \wedge x[j] \leq 0 \wedge t=t_0 \}\}$
    j := j+2
  fi
od
$\{\{ x=x' \wedge \max(ps,pt) \leq n+1 \wedge \forall a \in [1 \,..\, \min(ps,pt)-1], 2 \nmid a: x[a] \leq 0 \wedge (pt \leq n \rightarrow x[pt] > 0) \}\}$

P ≡ $x=x' \wedge \max(ps,pt) \leq n+1 \wedge j \leq \min(ps,pt) \wedge \forall a \in [1 \,..\, j-1], 2 \nmid a: x[a] \leq 0 \wedge (pt \leq n \rightarrow x[pt] > 0)$
t = n+1−i + pt

Megjegyzés

- Az interferencia-mentességhez elegendő csak azokat az u elemi utasításokat vizsgálni, amelyek közös változók értékeit módosítják.
- Elegendő továbbá csak azokat a pre_s feltételeket ellenőrizni, amelyek
  - a közös változókról szólnak,
  - nincsenek atomi utasássorozatba ágyazva

5. Példa: interferencia-mentesség

S_2: $\{\{ P \wedge x[j] > 0 \wedge t=t_0 \}\}$ pt := j

S_1:

$\{\{ x=x' \wedge i=1 \wedge ps=n+1 \}\}$
while $i<ps \leq pt \vee i<pt \leq ps$ **do**
  $\{\{ P \wedge t=t_0 \}\}$
  if $x[i] > 0$ **then**
    $\{\{ P \wedge x[i] > 0 \wedge t=t_0 \}\}$
    ps := i
  else
    $\{\{ P \wedge x[i] \leq 0 \wedge t=t_0 \}\}$
    i := i+2
  fi
od
$\{\{ x=x' \wedge \max(ps,pt) \leq n+1 \wedge \forall a \in [1 \,..\, \min(ps,pt)-1], 2 \text{ł} a: x[a] \leq 0 \wedge (ps \leq n \rightarrow x[ps]>0) \}\}$

$P \equiv x=x' \wedge \max(ps,pt) \leq n+1 \wedge i \leq \min(ps,pt) \wedge \forall a \in [1 \,..\, i-1], 2 \text{ł} a: x[a] \leq 0 \wedge (ps \leq n \rightarrow x[ps]>0)$
$t = n+1-i+ps$

5. Példa: interferencia-mentesség

S_1: $\{\{ P \wedge x[i] > 0 \wedge t=t_0 \}\}$ ps := i

S_2:

$\{\{ x=x' \wedge j=2 \wedge pt=n+1 \}\}$
while $j<ps \leq pt \vee j<pt \leq ps$ **do**
  $\{\{ P \wedge t=t_0 \}\}$
  if $x[j] > 0$ **then**
    $\{\{ P \wedge x[j] > 0 \wedge t=t_0 \}\}$
    pt := j
  else
    $\{\{ P \wedge x[j] \leq 0 \wedge t=t_0 \}\}$
    j := j+2
  fi
od
$\{\{ x=x' \wedge \max(ps,pt) \leq n+1 \wedge \forall a \in [2 \,..\, \min(ps,pt)-1], 2 \text{ł} a: x[a] \leq 0 \wedge (pt \leq n \rightarrow x[pt]>0) \}\}$

$P \equiv x=x' \wedge \max(ps,pt) \leq n+1 \wedge j \leq \min(ps,pt) \wedge \forall a \in [1 \,..\, j-1], 2 \text{ł} a: x[a] \leq 0 \wedge (pt \leq n \rightarrow x[pt]>0)$
$t = n+1-j+pt$