Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Departamento de Computación

Compilación de ejercicios resueltos

Primer Cuatrimestre 2020

Guia Práctica 5
Ejercicios entregables

Isaac Edgar Camacho Ocampo

Buenos Aires, 2020

Índice general

Capítulo 1

Ejercicio

Ejercicio 1. Sean $a, b, c, d \in \mathbb{N}$. Probar o refutar dando un contraejemplo las siguientes afirmaciones:

1. Si $a|b$ y $c|d$ entonces $ac|bd$. **2.** Si $a|b$ entonces $ac|bc$. **3.** Si $a \nmid bc$ entonces $a \nmid b$ y $a \nmid c$.

4. Si $ac|bc$ entonces $a|b$. **5.** Si $a|bc$ entonces $a|b$ o $a|c$. **6.** Si $a|c$ y $b|c$ entonces $ab|c$.

7. Si $4|a^2$ entonces $2|a$. **8.** Si $9|b + c$ entonces $9|b$ o $9|c$. **9.** Si $a|b + c^2$ entonces $a|b$.

Solución

1. si a divide a b y ademas c divide a d entonces existen enteros k y t que verifican

$$\begin{aligned}
a|b \Rightarrow &\ \text{existe k tal que } b = ka \\
c|d \Rightarrow &\ \text{existe t tal que } d = tc \\
\Rightarrow &\ \text{multiplicando ambas igualdades } bd = (ka) \cdot (tc) \\
\Rightarrow &\ \text{agrupando convenientemente } bd = (kt)ac \\
\Rightarrow &\ \text{pero } (kt)\epsilon\mathbb{Z} \text{ entonces } bd|ac
\end{aligned}$$

2.

$$\begin{aligned}
a|b \Rightarrow &\ \text{existe k tal que } b = ka \\
\Rightarrow &\ \text{multiplicando ambas igualdades por un entero c tenemos que } bc = kac \\
\Rightarrow &\ \text{agrupando convenientemente } bc = kac \\
\Rightarrow &\ \text{pero bc esta escrito como un entero por ac entonces } ac|bc
\end{aligned}$$

Capítulo 2

Demostración

Para demostrar la corrección del programa en SmallLang con respecto a la especificación, debemos probar que:

$$Post \Rightarrow wp(\ \textbf{código previo al ciclo}\ , Pc) \tag{2.1}$$

$$Pc \Rightarrow wp(\ \textbf{ciclo}, Qc) \tag{2.2}$$

$$Qc \Rightarrow wp(\ \textbf{código posterior al ciclo}\ , Post) \tag{2.3}$$

La parte 2.2 con el teoréma del invariante. Si probamos estas tres implicaciones, por el principio de monotonía sabemos que $Pre \Rightarrow wp(programacompleto, Post)$ y por lo tanto el programa es correcto, respecto a su especificación

2.1. Definición de Qc, Pc, I, fv

Primero vamos a definir los predicados que necesitamos para la demostracion.

$$
\begin{aligned}
Pre\quad & \{True\} \\
Post = \quad & \{r = True \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e)\} \\
Pc = \quad & (i = 0) \wedge (j = -1) \\
Qc = \quad & (j \neq -1) \leftrightarrow (\exists k : \mathbb{Z})(0 \leq k <\mid s \mid \ \wedge_L s[k] = e) \\
I = \quad & 0 \leq i \leq \mid s \mid) \wedge (j \neq -1) \leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i \rightarrow_L s[k] = e) \wedge (s[i] = e \rightarrow i = j) \\
fv = \quad & (\mid s \mid -i) \\
B = \quad & (i <\mid s \mid)
\end{aligned}
$$

2.1.1. Pre $\Rightarrow$ wp(código previo al ciclo, Pc)

$$
\begin{aligned}
Pre \rightarrow wp(i := 0; j := -1, Pc) \equiv \quad & wp(i := 0; wp(j := -1, Pc)) \\
\\
Calculamos\ wp(j := -1, Pc) & \\
wp(j := -1, Pc) \equiv \quad & def(-1) \wedge_L Pc^{j}_{-1} \\
\equiv \quad & wp(j := -1, ((i = 0) \wedge (j = -1))^{j}_{(-1)}) \\
\equiv \quad & def(-1) \wedge (i = 0) \wedge (-1 = -1) \\
E1 \equiv \quad & (i = 0) \\
\\
Calculamos\ wp(i := 0, E1) & \\
wp(i := 0, E1) \equiv \quad & def(0) \wedge_L E1^{i}_{0} \\
\equiv \quad & True \wedge_L (i = 0)^{i}_{0} \\
\equiv \quad & (0 = 0) \\
E2 \equiv \quad & True
\end{aligned}
$$

Por lo tanto demostramos que:

$$Pre \Rightarrow wp(\ \textbf{código previo al ciclo}\ , Pc)$$

2.1.2. Qc $\Rightarrow$ wp(if..then..else..fi, Post)

Recordamos la post

$$Post = \{r = True \leftrightarrow ((\exists k : \mathbb{Z})(0 \leq k < |s|) \wedge_L s[k] = e)\}$$

Calculamos:

$$
\begin{aligned}
wp(if.....endif, Post) &\equiv wp(if(j \neq -1)then\ r := True\ else\ r := False\ endif, Post) \\
&\equiv def(j \neq -1) \wedge_L [(j \neq -1) \wedge wp(r := True, Post)] \vee [(j = -1) \wedge wp(r := False, Post)] \\
&\equiv \underbrace{def(j \neq -1)}_{\text{True}} \wedge_L \quad [\qquad ” \qquad] \vee [\qquad ” \qquad] \\
&\equiv (j \leq -1 \wedge wp(r := True, Post)) \vee (j = -1 \wedge wp(r := False, Post)) \\
&\equiv (j \leq -1 \wedge {\color{red} wp(r := True, Post)}) \vee (j = -1 \wedge {\color{blue} wp(r := False, Post)})
\end{aligned}
$$

$$
\begin{aligned}
wp(r := True, Post) &\equiv def(True) \land_L Post^r_{true} \\
&\equiv (True = True) \leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < |s| \land_L s[k] = e) \\
wp(r := True, Post) &\equiv (\exists k : \mathbb{Z})(0 \leq k < |s| \land_L s[k] = e)
\end{aligned}
$$

$$
\begin{aligned}
wp(r := False, Post) &\equiv def(False) \land_L Post^r_{false} \\
&\equiv \underbrace{(false = True)}_{\text{False}} \leftrightarrow \underbrace{(\exists k : \mathbb{Z})(0 \leq k < |s| \land_L s[k] = e)}_{(*)}
\end{aligned}
$$

$(*)$ *entonces como* $(\exists k : \mathbb{Z})(0 \leq k < |s| \land_L s[k] = e)$ debe ser Falso, invierto desigualdad

$$
wp(r := False, Post) \equiv (\forall k : \mathbb{Z})(0 \leq k < |s| \land_L s[k] \neq e)
$$

$$
\begin{aligned}
wp(if...endif, Post) &\equiv (j \neq -1) \land (\exists k : \mathbb{Z})(0 \leq k < |s| \land_L s[k] = e) \lor \\
&\quad (j = -1) \land (\forall k : \mathbb{Z})(0 \leq k < |s| \land_L s[k] \neq e) \\
&\quad (aplico\ (p \land q) \lor (\neg p \land \neg q) \equiv p \leftrightarrow q) \\
E3 &\equiv (j \neq -1) \leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < |s| \land_L s[k] = e)
\end{aligned}
$$

Chequeamos $Qc \rightarrow E3$

$$
Qc \equiv j \neq -1 \leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < |s| \land_L s[k] = e) \equiv E3
$$

Por lo tanto demostramos que:

$$
Qc \Rightarrow E3
$$

Capítulo 3

Demostración de la corrección del ciclo

3.0.1. Qc $\Rightarrow$ wp(ciclo, Qc)

Teorema. Sean un predicado I y una función $fv : \mathbb{V} \to \mathbb{Z}$ (donde $\mathbb{V}$ es el producto cartesiano de los dominios de las variables del programa), y supongamos que I $\to$ def(B). Si se cumplen:

1) $PC \Rightarrow I$,
2) $\{I \wedge B\}S\{I\}$,
3) $I \wedge \neg B \Rightarrow Qc$,
4) $\{I \wedge B \wedge v_0 = fv\}S\{fv < v_0\}$,
5) $I \wedge fv \leq 0 \Rightarrow \neg B$,

... entonces la siguiente tripla de Hoare es válida: {PC } while B do S endwhile {QC }

3.0.2. $Pc \ \rightarrow I$

$$\begin{aligned}
Pc = \quad & (i = 0) \wedge (j = -1) \\
I = \quad & 0 \leq i \leq \mid s \mid) \wedge (j \neq -1) \leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i \rightarrow_L s[k] = e) \wedge (s[i] = e \rightarrow i = j) \\
\equiv \quad & (i = 0) \wedge (j = -1) \rightarrow (0 \leq i \leq \mid s \mid) \checkmark (\mathit{trivial}) \\
\equiv \quad & (j \neq -1 \wedge (\exists k : \mathbb{Z})(0 \leq k < i \wedge_L s[k] = e)) \vee (j = -1 \wedge (\forall k : \mathbb{Z})(0 \leq k < i \wedge_L s[k] \neq e))\checkmark
\end{aligned}$$

(porque se cumple trivialmente que $(j = -1)$ y por vacuidad $(j = -1 \wedge (\forall k : \mathbb{Z})(0 \leq k < i \wedge_L s[k] \neq e))$ ya que no hay ningún número que sea mayor o igual a cero y menor a cero a la vez)

3.0.3. $(I \ \wedge \neg B) \rightarrow Qc$

$$Q_c \equiv (j \neq -1) \Leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < |s| \wedge_L s[k] = e)$$
$$B \equiv (i < |s|)$$
$$I \equiv 0 \leq i \leq |s| \wedge j \neq -1 \Leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i \Rightarrow_L s[k] = e) \wedge (s[i] = e \rightarrow i = j)$$

Q_c✓

(porque $I \wedge \neg B$ implica que $i = |s|$ ya que por I, $i \leq |s|$ y por $\neg B$ $i \geq |s|$ y si aplico esto a I me queda Q_c)

3.0.4. $\{I \wedge B\}$ ciclo $\{I\}$

$$\begin{aligned} I = & \quad (0 \leq i \leq\mid s \mid) \wedge (j \neq -1) \leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i \rightarrow_L s[k] = e) \wedge (s[i] = e \rightarrow i = j) \\ B \equiv & \quad (i < |s|) \end{aligned}$$

Ciclo

$if(s[i] = e)then$

$\quad j := i$

$else$

$\quad skip$

$endif$

$i := i + 1$

Veamos si $(I \wedge B) \Rightarrow wp(\text{if...then..else..fi}, \ i := i + 1, I)$

$$wp(if..fi, i := i + 1, I) \equiv wp(if..fi, \underbrace{wp(i := i + 1, I)}_{debo\ calcular})$$

$$\begin{aligned} wp(i := i + 1, I) \equiv & \ \underbrace{def(i + 1)}_{True} \wedge_L I^i_{i+1} \\ \equiv & \ (0 \leq i + 1 \leq |s|) \wedge_L (j \neq -1 \Leftrightarrow (\exists k : \mathbb{Z}))(0 \leq k < i + 1 \Rightarrow_L s[k] = e)) \wedge \\ & \ (s[i + 1] = e \rightarrow i + 1 = j) \end{aligned}$$

$$\begin{aligned} E4 \equiv & \ (0 \leq i + 1 \leq |s|) \wedge_L (j \neq -1 \Leftrightarrow (\exists k : \mathbb{Z}))(0 \leq k < i + 1 \Rightarrow_L s[k] = e)) \wedge \\ & \ (s[i + 1] = e \rightarrow i + 1 = j) \end{aligned}$$

Calculo $wp(if...then..else..fi, E4)$

$$\begin{aligned}
wp(if..fi, E4) &\equiv \underbrace{def(s[i]=e)}_{True} \wedge_L \ [\ s[i]=e \wedge wp(j:=i, E4)\ ] \vee [\ s[i] \neq e \wedge wp(skip, E_4)] \\
&\equiv (s[i]=e \wedge \underbrace{def(i)}_{True} \wedge E4^j_i) \vee (s[i] \neq e \wedge E4) \\
&\equiv (s[i]=e \wedge E4^j_i) \vee (s[i] \neq e \wedge E4) \\
&\equiv (s[i]=e) \quad \wedge \\
&\quad (0 \leq i+1 \leq |s|) \wedge_L (i \neq -1 \Leftrightarrow (\exists k:\mathbb{Z}))(0 \leq k < i+1 \Rightarrow_L s[k]=e))\wedge \\
&\quad (s[i+1]=e \rightarrow i+1=i) \qquad \vee \\
&\quad (s[i] \neq e) \quad \wedge \\
&\quad (0 \leq i+1 \leq |s| \wedge_L (j \neq -1 \Leftrightarrow (\exists k:\mathbb{Z})(0 \leq k \leq i+1 \Rightarrow_L s[k]=e))\wedge \\
&\quad (s[i+1]=e \rightarrow i+1=j)
\end{aligned}$$

$$\begin{aligned}
E5 \equiv\ & (0 \leq i+1 \leq |s|)\wedge_L \\
& (s[i]=e \wedge (i \neq -1 \Leftrightarrow (\exists k:\mathbb{Z})(0 \leq k < i+1 \Rightarrow_L s[k]=e))\wedge \\
& (s[i+1]=e \rightarrow i+1=i)) \\
& \vee \\
& (s[i] \neq e \wedge (j \neq -1 \Leftrightarrow (\exists k:\mathbb{Z})(0 \leq k \leq i+1 \Rightarrow_L s[k]=e))\wedge \\
& (s[i+1]=e \rightarrow i+1=j))
\end{aligned}$$

Segun el invariante $(s[i]=e \rightarrow i=j)$, y usando la equivalencia $(p \wedge q) \vee (\neg p \wedge q) \equiv q$

$$\begin{aligned}
E5 \equiv\ & (0 \leq i+1 \leq |s|)\wedge_L \\
& (\underbrace{s[i]=e}_{\text{por I}} \wedge (\underbrace{i \neq -1}_{\text{i=j}} \Leftrightarrow (\exists k:\mathbb{Z})(0 \leq k < i+1 \Rightarrow_L s[k]=e))\wedge \\
& (s[i+1]=e \rightarrow \underbrace{i+1=i}_{i=j})) \\
& \vee \\
& (s[i] \neq e \wedge (j \neq -1 \Leftrightarrow (\exists k:\mathbb{Z})(0 \leq k \leq i+1 \Rightarrow_L s[k]=e))\wedge \\
& (s[i+1]=e \rightarrow i+1=j))
\end{aligned}$$

$$\begin{aligned}
E5 \equiv\ & (0 \leq i+1 \leq |s|)\wedge_L \\
& (s[i]=e \wedge (j \neq -1 \Leftrightarrow (\exists k:\mathbb{Z})(0 \leq k < i+1 \Rightarrow_L s[k]=e))\wedge \\
& (s[i+1]=e \rightarrow i+1=j)) \\
& \vee \\
& (s[i] \neq e \wedge (j \neq -1 \Leftrightarrow (\exists k:\mathbb{Z})(0 \leq k \leq i+1 \Rightarrow_L s[k]=e))\wedge \\
& (s[i+1]=e \rightarrow i+1=j))
\end{aligned}$$

$$\begin{aligned} & usando (p \wedge q) \vee (\neg p \wedge q) \equiv q \\ E5 \equiv & (0 \leq i+1 \leq |s|) \wedge_L \\ & (j \neq -1 \Leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i+1 \Rightarrow_L s[k] = e) \wedge \\ & (s[i+1] = e \rightarrow i+1 = j) \end{aligned}$$

Ahora veamos si $\{I \wedge B\} \Rightarrow E5$

Hipótesis:

1. $B \equiv \quad (i < |s|)$
2. $I = \quad (0 \leq i \leq| s |)$
3. $\quad (j \neq -1) \leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i \rightarrow_L s[k] = e)$
4. $\quad (s[i] = e \rightarrow i = j)$

Tésis

1. $(0 \leq i+1 \leq |s|)$
2. $(j \neq -1 \Leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i+1 \Rightarrow_L s[k] = e)$
3. $(s[i+1] = e \rightarrow i+1 = j)$

Comenzamos por la tésis 1 $(0 \leq i+1 \leq |s|)$

$$\begin{array}{lcl} \textbf{segun la hip 2} & \Rightarrow & 0 \leq i \\ & \Rightarrow & 0 \leq i+1 \\ \textbf{segun la hip 1} & \Rightarrow & i < |s| \\ & \Rightarrow & i+1 < |s|+1 \\ & \Rightarrow & i+1 \leq |s| \end{array}$$

Por lo tanto se cumple la tésis 1 $(0 \leq i+1 \leq |s|)$

Continuamos por la tésis 2

$$(j \neq -1 \Leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i+1 \Rightarrow_L s[k] = e)$$

$$
\begin{aligned}
0 \leq k < i+1 \quad &\equiv \quad 0 \leq k \leq i \\
\textbf{sabemos que} \quad i < |s| \quad &\equiv \quad 0 \leq k \leq i < |s| \\
(*) &\equiv \quad 0 \leq k < |s| \\
(j \neq -1 \Leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i+1 \Rightarrow_L s[k] = e) &\equiv (j \neq -1 \Leftrightarrow (\exists k : \mathbb{Z})(\underbrace{0 \leq k < |s|}_{(*)} \Rightarrow_L s[k] = e) \\
&\equiv (j \neq -1 \Leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < |s| \Rightarrow_L s[k] = e)
\end{aligned}
$$

Esto significa que j sera distinto de -1 cuando exista un k que estando en rango verifique que el elemento en la posicion k sea e, y sera falso cuando e no este en la secuencia.

Continuamos por la tésis 3

$$(s[i+1] = e \leftrightarrow i+1 = j)$$

$$
\begin{aligned}
\textbf{segun la hip 4} \quad i = j \quad &\leftrightarrow \quad s[i] = e \\
&\leftrightarrow \quad s[j] = e \\
\textbf{pero si} \quad j = i+1 \quad &\leftrightarrow \quad s[i+1] = e \\
\textbf{de igual forma si} \quad s[i] = e \quad &\leftrightarrow \quad j = i \\
s[i+1] = e \quad &\leftrightarrow \quad j = i+1
\end{aligned}
$$

Esto dice que si e esta en la posicion i+1, entonces j debe ser i+1, ya que j era igual a i.

Por lo tanto se cumple

$$(s[i+1] = e \leftrightarrow i+1 = j)$$

3.0.5. $I \wedge f_v \leq 0 \rightarrow \neg B$

Queremos demostrar que cuando la función variante llega a un valor determinado entonces se termina el ciclo o dicho de otra forma la guarda se hace falsa.

Hipótesis

$$
\begin{aligned}
I = \quad & (0 \leq i \leq \mid s \mid) \wedge (j \neq -1) \leftrightarrow (\exists k : \mathbb{Z})(0 \leq k < i \rightarrow_L s[k] = e) \wedge (s[i] = e \rightarrow i = j) \\
fv = \quad & (\mid s \mid -i) \leq 0
\end{aligned}
$$

Tésis

$$\neg B \equiv \quad (i \geq |s|)$$

Demostración

$$
\begin{aligned}
fv \equiv \quad & \mid s \mid -i \leq 0 \\
\equiv \quad & \mid s \mid \leq i \\
\equiv \quad & (i \geq |s|) \qquad (\textbf{ que es } \neg B)
\end{aligned}
$$

Por lo tanto se cumple la tésis $(I \wedge f_v \leq 0) \rightarrow \neg B$

3.0.6. $\{I \wedge B \wedge v_0 = fv\}S\{fv < v_0\}$

Finalemente vamos a demostrar que la funcion variante decrece, esto equivale a decir que si estamos al principio del ciclo en donde valen, tanto el invariante, la guarda y donde la funcion variante toma cierto valor v_0, despues de ejecutar el cuerpo del ciclo la fv va a tomar un valor estrictamente menor a v_0 .

Ciclo

$\mathbf{S1} if(s[i] = e) then$

$j := i$

$else$

$skip$

$endif$

$\mathbf{S2} i := i + 1$

Si llamamos **S1 al if..endif** y **S2 a i:=i+1**. Debemos hallar $wp(S1; S2, fv < v_0)$

$$wp(S1; S2, fv < v_0) \equiv wp(S1,\ wp(S2, |s| - i < v_0))$$

$$\begin{aligned}
wp(S2, |s| - i < v_0) &\equiv wp(i := i + 1, |s| - i < v_0) \\
&\equiv \underbrace{def(i+1)}_{True} \wedge_L (|s| - i < v_0)^i_{i+1} \\
&\equiv (|s| - i < v_0)^i_{i+1} \\
E2 \equiv wp(S2, |s| - i < v_0) &\equiv (|s| - (i + 1) < v_0)
\end{aligned}$$

Calculamos la precondición más débil del if..fi respecto de E2.

$$\begin{aligned}
wp(S1, E2) &\equiv wp(if..fi, E2) \\
&\equiv \underbrace{def(s[i] = e)}_{True} \wedge_L\ [\ s[i] = e \wedge wp(j := i, E2)\ ] \vee [\ s[i] \neq e \wedge wp(skip, E2)] \\
&\equiv [\ s[i] = e \wedge wp(j := i, E2)\ ] \vee [\ s[i] \neq e \wedge E2] \\
&\equiv (\ s[i] = e \wedge (def(i) \wedge_L E2^j_i)\ ) \vee (\ s[i] \neq e \wedge E2) \\
&\equiv (\ s[i] = e \wedge (E2^j_i)\ ) \vee (\ s[i] \neq e \wedge E2) \\
&\equiv (\ s[i] = e \wedge (|s| - (i + 1) < v_0)\ ) \vee (\ s[i] \neq e \wedge (|s| - (i + 1) < v_0))
\end{aligned}$$

$$usando (p \wedge q) \vee (\neg p \wedge q) \equiv q$$

$$E1 \equiv (|s| - (i + 1) < v_0)$$

$$E1 \equiv (|s| - i - 1 < v_0)$$

Por lo tanto la precondición más débil que queriamos calular es E1, ahora debemos verificar que

$$\{I \land B \land v_0 = fv\} \Rightarrow E1$$

$$\begin{aligned}
fv &= v_0 \\
|s| - i &= v_0 \qquad \text{(restando 1 a ambos miembros obtenemos)} \\
|s| - i - 1 &= v_0 - 1 \qquad \text{(pero } (v_0 - 1) < v_0\text{)} \\
|s| - i - 1 &= v_0 - 1 < v_0 \\
|s| - i - 1 &< v_0
\end{aligned}$$

Por lo tanto nuestra funcion variante decrece y

$$\{I \land B \land v_0 = fv\} \Rightarrow E1$$

Capítulo 4

Conclusiones

Estamos en condiciones de afirmar que nuestro programa, habiendo demostrado a través del teorema de invarinate que el ciclo es correcto respecto de su espeficicación y a través del teorema de terminacion del ciclo, que el mismo finaliza siempre y ademas termina en un estado en que vale la postcondicion del ciclo.
Con todo esto probado y por el principio de monotonia probamos que el programa completo es correcto respecto de la especificación dada.