Theorem Problem

SNU 4541.664A Program Analysis

Spring 2006

Note 10-2

Prof. Kwangkeun Yi

요약해석 디자인과 구현의 예

프로그램의 요약해석: 방정식 풀기

명령형 언어 프로그램의 요약해석

$$
\begin{array}{rcl}
C & \to & \texttt{skip} \mid x \texttt{ := } E \mid C \texttt{ ; } C \\
& \mid & \texttt{if } B\ C\ C \\
& \mid & \texttt{while } B\ C \\
E & \to & n \quad (n \in \mathbb{Z}) \mid x \\
& \mid & E \texttt{ + } E \mid B \quad \text{(boolean expr)}
\end{array}
$$

의미공간은

$$
\begin{array}{rcl}
\mathcal{C}\ C & \in & 2^{Memory} \to 2^{Memory} \\
\mathcal{V}\ E & \in & 2^{Memory} \to 2^{Value} \\
\mathcal{B}\ B & \in & 2^{Memory} \to 2^{Memory} \\
Memory & = & Loc \xrightarrow{\text{fin}} Value \\
Value & = & \mathbb{Z} + \mathbb{B} \\
Loc & = & Var \\
\mathbb{B} & = & \{T, F\}
\end{array}
$$

세 갈래 길

- 프로그램 C의 의미를 조립식으로 (최소고정점) (10-1.pdf)
- 프로그램 C의 의미를 기계상태의 전이과정으로 (최소고정점) (10-1.pdf)
- 프로그램 C의 의미를 방정식의 해로 (최소고정점)

방정식의 해로 바라보기: "숲"을 보는 시각

이 시각이 필요한 이유

- 대상 언어가 $\texttt{while}\ B\ C$ 이외의 방식으로 반복을 표현할 수 있으면?
  - goto, exception raise/handle, recursive call
- 반복의 내용에 해당하는 의미 함수

$$\mathcal{C}\ (\texttt{while}\ B\ C)\ M = \mathcal{B}\ \neg B\ (\mathit{fix} \lambda X.M \cup \mathcal{C}\ C\ (\mathcal{B}\ B\ X))$$

를 "나무"만 봐서는 알아내기 힘듬.

- $\mathcal{C}\ (\texttt{goto}\ L)\ M = \mathit{fix} \lambda X.?$
- $\mathcal{C}\ (\texttt{raise DivByZero})\ M = \mathit{fix} \lambda X.?$
- $\mathcal{C}\ (f\ E)\ M = \mathit{fix} \lambda X.?$

C의 의미를 방정식의 해로

$$m \in Memory \qquad M \in 2^{Memory}$$

$$\begin{aligned}
\mathcal{C}\,\texttt{skip}\,M &= M \\
\mathcal{C}\,(x\ \texttt{:=}\ E)\,M &= \{m\{x \mapsto v\} \mid m \in M, v \in \mathcal{V}\,E\,M\} \\
\mathcal{C}\,(C_1\ ;\ C_2)\,M &= \mathcal{C}\,C_2\,(\mathcal{C}\,C_1\,M) \\
\mathcal{C}\,(\texttt{if}\ B\ C_1\ C_2)\,M &= \mathcal{C}\,C_1\,(\mathcal{B}\,B\,M)\ \cup\ \mathcal{C}\,C_2\,(\mathcal{B}\,\neg B\,M) \\
\mathcal{C}\,(\texttt{while}\ B\ C)\,M &= \mathcal{B}\,\neg B\,(M\ \cup\ (\mathcal{C}\,(\texttt{while}\ B\ C)\,(\mathcal{C}\,C\,(\mathcal{B}\,B\,M)))) \\
\mathcal{V}\,n\,M &= \{n\} \\
\mathcal{V}\,x\,M &= \{m\,x \mid m \in M\} \\
\mathcal{V}\,(E_1\ \texttt{+}\ E_2)\,M &= \{v_1 + v_2 \mid v_1 \in \mathcal{V}\,E_1\,M, v_2 \in \mathcal{V}\,E_2\,M\} \\
\mathcal{B}\,B\,M &= \cup\{M' \mid \mathcal{V}\,B\,M' = \{T\}, M' \subseteq M\}
\end{aligned}$$

정의 vs 방정식

프로그램 C의 의미는 $\mathcal{C}\,C$로 정의되는가? No.

while-문의 경우:

$$\begin{aligned}\mathcal{C}\,(\texttt{while}\ B\ C) &= \cdots\mathcal{C}\,(\texttt{while}\ B\ C)\cdots \\ &= \cdots(\cdots\mathcal{C}\,(\texttt{while}\ B\ C)\cdots)\cdots \\ &= \cdots\end{aligned}$$

답) $\mathcal{C}\,C$는 명령문 프로그램 C를 가지고 만드는 방정식을 표현하는 것 뿐. 그 방정식의 해가 프로그램 C의 의미.

프로그램 C의 의미 방정식

$$\mathcal{C}\,C = \cdots$$

의 오른편은 위에서 표현한 함수 $\mathcal{C}$의 내용을 고스란히 가지는 상위의 함수

$$\mathcal{F} : (Cmd \to 2^{Memory} \to 2^{Memory}) \to (Cmd \to 2^{Memory} \to 2^{Memory})$$

가 정의하는 $\mathcal{F}\,\mathcal{C}\,C$가 된다:

$$\begin{array}{rcl}
\mathcal{F}\,\mathcal{C}\,\texttt{skip}\,M & = & M \\
\mathcal{F}\,\mathcal{C}\,(x\ \texttt{:=}\ E)\,M & = & \{m\{x \mapsto v\} \mid m \in M, v \in \mathcal{V}\,E\,M\} \\
\mathcal{F}\,\mathcal{C}\,(C_1\ ;\ C_2)\,M & = & \mathcal{C}\,C_2\,(\mathcal{C}\,C_1\,M) \\
\mathcal{F}\,\mathcal{C}\,(\texttt{if}\ B\ C_1\ C_2)\,M & = & \mathcal{C}\,C_1\,(\mathcal{B}\,B\,M)\ \cup\ \mathcal{C}\,C_2\,(\mathcal{B}\,\neg B\,M) \\
\mathcal{F}\,\mathcal{C}\,(\texttt{while}\ B\ C)\,M & = & \mathcal{B}\,\neg B\,(M\ \cup\ (\mathcal{C}\,(\texttt{while}\ B\ C)\,(\mathcal{C}\,C\,(\mathcal{B}\,B\,M))))
\end{array}$$

C의 의미 방정식

프로그램 C의 의미는 그 의미 $\mathcal{C}\,C$에 대한 방정식

$$\mathcal{C}\,C = \mathcal{F}\,\mathcal{C}\,C$$

와 C안의 모든 명령문 C_i들에 대한 방정식

$$\mathcal{C}\,C_i = \mathcal{F}\,\mathcal{C}\,C_i$$

들의 해를 가지고 정의된다.

그러한 $\mathcal{F}$를 통해서 프로그램 C로 부터 도출되는 연립방정식을 하나의 함수 $\mathcal{F}_C$를 가지고 표현하면

$$\begin{pmatrix} X_0 \\ \vdots \\ X_n \end{pmatrix} = \mathcal{F}_C \begin{pmatrix} X_0 \\ \vdots \\ X_n \end{pmatrix}$$

이 되고, 방정식의 주인공 X_i는 $\mathcal{C}\, C_i$를 대신에 쓴 것. (C안의 명령문들의 갯수는 n)

- $\mathcal{F}_C$의 정의는 $\mathcal{F}$와 $\mathcal{V}$ 로 부터 명백
- 방정식의 해는 $\mathcal{F}_C$의 최소고정점
- $\mathcal{F}_C$의 최소고정점은 존재: (why?)

$$lfp\mathcal{F}_C = \bigsqcup_i (\mathcal{F}_C^i \langle \bot, \cdots, \bot \rangle)$$

$lfp\mathcal{F}_C$는 $lfp\mathcal{F}$의 일부

방정식의 해

$$lfp\mathcal{F}_C$$

는

$$lfp\mathcal{F} \in Cmd \to 2^{Memory} \to 2^{Memory}$$

중에서 프로그램 C를 구성하는 명령문들의 의미들로 구성된다:

$$lfp\mathcal{F}_C = \langle (lfp\mathcal{F})C_0, (lfp\mathcal{F})C_1, \cdots, (lfp\mathcal{F})C_n \rangle$$

- $\mathcal{F}_C$를 구성하는 속 내용은 모두 $\mathcal{F}$의 정의 그대로.
- $lfp\mathcal{F} \in Cmd \to 2^{Memory} \to 2^{Memory}$는 모든 명령문에 대한 의미.

앞으로

- 연속인(*continuous*) 요약 의미 함수를 정의:

  $\hat{\mathcal{F}} \in (Cmd \to \hat{Memory} \to \hat{Memory}) \to (Cmd \to \hat{Memory} \to \hat{Memory})$

- 프로그램 C의 요약 의미는 $\hat{\mathcal{F}}$로 부터 도출되는 연립방정식

$$\begin{pmatrix} X_0 \\ \vdots \\ X_n \end{pmatrix} = \hat{\mathcal{F}}_C \begin{pmatrix} X_0 \\ \vdots \\ X_n \end{pmatrix}$$

  의 해, 즉 $lfp\hat{\mathcal{F}}_C$.

- 방정식의 해 $lfp\hat{\mathcal{F}}_C$는 $lfp\mathcal{F}$의 일부:

$$lfp\hat{\mathcal{F}}_C = \langle (lfp\hat{\mathcal{F}})\, C_0, (lfp\hat{\mathcal{F}})\, C_1, \cdots, (lfp\hat{\mathcal{F}})\, C_n \rangle$$

- 올바른가 검증:

$$\alpha(lfp\mathcal{F}) \sqsubseteq lfp\hat{\mathcal{F}}?$$

  즉, 요약해석의 틀에 의해서 다음을 증명하면 됨:

$$\alpha \circ \mathcal{F} \sqsubseteq \hat{\mathcal{F}} \circ \alpha \quad \text{혹은} \quad \alpha(f) \sqsubseteq g \implies \mathcal{F}\, f \sqsubseteq \hat{\mathcal{F}}\, g$$

요약된 의미 방정식

프로그램 C의 요약된 의미는 다음의 요약공간

$$\begin{aligned} \hat{\mathcal{C}}\,C &\in \hat{Memory} \to \hat{Memory} \\ \hat{\mathcal{V}}\,E &\in \hat{Memory} \to \hat{Value} \\ \hat{\mathcal{B}}\,B &\in \hat{Memory} \to \hat{Memory} \end{aligned}$$

갈로아 연결된 요약공간

$$2^{Memory} \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} \hat{Memory} \qquad 2^{Value} \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} \hat{Value}$$

요약 의미 함수 1

$$
\begin{array}{rcl}
\hat{\mathcal{C}}\,\texttt{skip}\,\hat{m} & = & \hat{m} \\
\hat{\mathcal{C}}\,(x\ \texttt{:=}\ E)\,\hat{m} & = & \alpha_1\{m\{x \mapsto v\} \mid m \in \gamma_1\hat{m}, v \in \gamma_2(\hat{\mathcal{V}}\,E\,\hat{m})\} \\
\hat{\mathcal{C}}\,(C_1\ ;\ C_2)\,\hat{m} & = & \hat{\mathcal{C}}\,C_2\,(\hat{\mathcal{C}}\,C_1\,\hat{m}) \\
\hat{\mathcal{C}}\,(\texttt{if}\ B\ C_1\ C_2)\,\hat{m} & = & \hat{\mathcal{C}}\,C_1\,(\hat{\mathcal{B}}\,B\,\hat{m}) \sqcup \hat{\mathcal{C}}\,C_1\,(\hat{\mathcal{B}}\,\neg B\,\hat{m}) \\
\hat{\mathcal{C}}\,(\texttt{while}\ B\ C)\,\hat{m} & = & \hat{\mathcal{B}}\,\neg B\,(\hat{m} \sqcup \hat{\mathcal{C}}\,(\texttt{while}\ B\ C)\,(\hat{\mathcal{B}}\,B\,\hat{m})) \\
\hat{\mathcal{V}}\,n\,\hat{m} & = & \alpha_2\{n\} \\
\hat{\mathcal{V}}\,x\,\hat{m} & = & \hat{m}\,x \\
\hat{\mathcal{V}}\,(E_1\ \texttt{+}\ E_2)\,m & = & \alpha_2\{v_1 + v_2 \mid v_1 \in \gamma_2(\hat{\mathcal{V}}\,E_1\,\hat{m}), v_2 \in \gamma_2(\hat{\mathcal{V}}\,E_2\,\hat{m})\} \\
\hat{\mathcal{B}}\,B\,\hat{m} & = & \alpha_1(\cup\{M' \mid \mathcal{V}\,B\,M' = \{T\}, \hat{M}' \subseteq \gamma_1\hat{m}\})
\end{array}
$$

명령문 C의 요약해석 방정식

$$\hat{\mathcal{C}}\, C = \cdots$$

의 오른편은 위에서 표현한 재귀함수 $\hat{\mathcal{C}}$의 내용을 고스란히 가지는 상위의 함수

$$\hat{\mathcal{F}} : (Cmd \to \hat{Memory} \to \hat{Memory}) \to (Cmd \to \hat{Memory} \to \hat{Memory})$$

가 정의하는 $\hat{\mathcal{F}}\,\hat{\mathcal{C}}\,C$가 되겠다:

$$\begin{array}{rcl}
\hat{\mathcal{F}}\,\hat{\mathcal{C}}\,\texttt{skip}\,\hat{m} & = & \hat{m} \\
\hat{\mathcal{F}}\,\hat{\mathcal{C}}\,(x\ \texttt{:=}\ E)\,\hat{m} & = & \alpha_1\{m\{x \mapsto v\} \mid m \in \gamma_1\hat{m}, v \in \gamma_2(\hat{\mathcal{V}}\ E\ \hat{m})\} \\
\hat{\mathcal{F}}\,\hat{\mathcal{C}}\,(C_1\ ;\ C_2)\,\hat{m} & = & \hat{\mathcal{C}}\,C_2\,(\hat{\mathcal{C}}\,C_1\,\hat{m}) \\
\hat{\mathcal{F}}\,\hat{\mathcal{C}}\,(\texttt{if}\ B\ C_1\ C_2)\,\hat{m} & = & \hat{\mathcal{C}}\,C_1\,(\hat{\mathcal{B}}\,B\,\hat{m}) \sqcup \hat{\mathcal{C}}\,C_1\,(\hat{\mathcal{B}}\,\neg B\,\hat{m}) \\
\hat{\mathcal{F}}\,\hat{\mathcal{C}}\,(\texttt{while}\ B\ C)\,\hat{m} & = & \hat{\mathcal{B}}\,\neg B\,(\hat{m} \sqcup \hat{\mathcal{C}}\,(\texttt{while}\ B\ C)\,(\hat{\mathcal{B}}\,B\,\hat{m}))
\end{array}$$

요약해석 틀에 의해, $lfp\hat{\mathcal{F}}$이 $lfp\mathcal{F}$의 안전한 요약이려면, 증명할 것은 아래 둘 중 하나이다:

- $\alpha \circ \mathcal{F} \sqsubseteq \hat{\mathcal{F}} \circ \alpha$, 즉, 모든 프로그램 C에 대해서

$$(\alpha(\mathcal{F}\ f))\ C \sqsubseteq (\hat{\mathcal{F}}(\alpha\ f))\ C,$$

  혹은,

- $\alpha\ f \sqsubseteq \hat{f}$ 이면 $\alpha(\mathcal{F}\ f) \sqsubseteq \hat{\mathcal{F}}\ \hat{f}$, 즉, 모든 프로그램 C에 대해서

$$(\alpha(\mathcal{F}\ f))\ C \sqsubseteq (\hat{\mathcal{F}}\ \hat{f})\ C.$$

구현

분석의 구현은, 분석할 프로그램 C가 주어졌을 때 연립방정식

$$\begin{pmatrix} X_0 \\ \vdots \\ X_n \end{pmatrix} = \hat{\mathcal{F}}_C \begin{pmatrix} X_0 \\ \vdots \\ X_n \end{pmatrix}$$

을 풀면된다. 연립방정식은 C안에 있는 모든 명령문들 C_i와 식들 E_i의 요약 의미

$$\hat{\mathcal{C}}\,C_i \in \hat{Memory} \to \hat{Memory}$$
$$\hat{\mathcal{V}}\,E_i \in \hat{Memory} \to \hat{Value}$$

에 대한 방정식.

방정식의 해(프로그램의 요약된 의미)는 메모리에서 메모리로 가는 함수가 된다.

요약 의미 함수 2

앞으로의 예에서는 다음의 것을 사용:

$$
\begin{aligned}
\hat{\mathcal{C}}\,\texttt{skip}\,\hat{m} &= \hat{m} \\
\hat{\mathcal{C}}\,(x\ \texttt{:=}\ E)\,\hat{m} &= \hat{m}\{x \mapsto \hat{\mathcal{V}}\,E\,\hat{m}\} \\
\hat{\mathcal{C}}\,(\texttt{if}\ E\ C_1\ C_2)\,\hat{m} &= (\hat{\mathcal{C}}\,C_1\,\hat{m}) \sqcup (\hat{\mathcal{C}}\,C_2\,\hat{m}) \\
\hat{\mathcal{C}}\,(C_1\ ;\ C_2)\,\hat{m} &= \hat{\mathcal{C}}\,C_2(\hat{\mathcal{C}}\,C_1\,\hat{m}) \\
\hat{\mathcal{C}}\,(\texttt{while}\ E\ C)\,\hat{m} &= \hat{m} \sqcup (\hat{\mathcal{C}}\,(\texttt{while}\ E\ C)\,(\hat{\mathcal{C}}\,C\,\hat{m})) \\
\hat{\mathcal{V}}\,n\,\hat{m} &= \alpha\{n\} \\
\hat{\mathcal{V}}\,x\,\hat{m} &= \hat{m}\,x \\
\hat{\mathcal{V}}\,(E_1\ \texttt{+}\ E_2)\,\hat{m} &= (\hat{\mathcal{V}}\,E_1\,\hat{m}) \hat{+} (\hat{\mathcal{V}}\,E_2\,\hat{m})
\end{aligned}
$$

예

다음 프로그램

$$x := 1 \,;\, \texttt{while}\,(0 < x)\; x := x + 1$$

을 생각하자. 각 부품마다 번호를 붙이자.

$$\underbrace{\underbrace{x := 1}_{1} \,;\, \texttt{while}\,(0 < x)\; \underbrace{x := \underbrace{x + 1}_{4}}_{3}}_{2}{}_{0}$$

각 부품의 의미 $\hat{\mathcal{C}}\, C_i$와 $\hat{\mathcal{V}}\, E_i$에 대한 방정식은 각각 $\hat{\mathcal{F}}\, C_i$와 $\hat{\mathcal{V}}\, E_i$에 의해서 아래와 같이 정의된다:

$$\begin{aligned}
\hat{\mathcal{C}}\, C_0 &= \lambda \hat{m}.\hat{\mathcal{C}}\, C_2(\hat{\mathcal{C}}\, C_1\, \hat{m}) \\
\hat{\mathcal{C}}\, C_1 &= \lambda \hat{m}.\hat{m}\{x \mapsto \hat{\mathcal{V}}\, 1\, \hat{m}\} \\
&= \lambda \hat{m}.\hat{m}\{x \mapsto \alpha\{1\}\} \\
\hat{\mathcal{C}}\, C_2 &= \lambda \hat{m}.\hat{m} \sqcup (\hat{\mathcal{C}}\, C_2\, (\hat{\mathcal{C}}\, C_3\, \hat{m})) \\
\hat{\mathcal{C}}\, C_3 &= \lambda \hat{m}.\hat{m}\{x \mapsto \hat{\mathcal{V}}\, E_4\, \hat{m}\} \\
&= \lambda \hat{m}.\hat{m}\{x \mapsto (\hat{m}\, x) \hat{+} \alpha\{1\}\}
\end{aligned}$$

방정식의 주인공 $\hat{\mathcal{C}}\, C_i$를 $\hat{X}_i$로 해서 다시 쓰면,

$$\begin{aligned}
\hat{X}_0 &= \lambda \hat{m}.\hat{X}_2(\hat{X}_1\, \hat{m}) \\
\hat{X}_1 &= \lambda \hat{m}.\hat{m}\{x \mapsto \alpha\{1\}\} \\
\hat{X}_2 &= \lambda \hat{m}.\hat{m} \sqcup (\hat{X}_2\, (\hat{X}_3\, \hat{m})) \\
\hat{X}_3 &= \lambda \hat{m}.\hat{m}\{x \mapsto (\hat{m}\, x) \hat{+} \alpha\{1\}\}
\end{aligned}$$

$2^{\mathbb{Z}} \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \hat{\mathbb{Z}}$의 두가지 경우를 살피자

- $\hat{\mathbb{Z}}$의 높이가 유한한 경우
- $\hat{\mathbb{Z}}$의 높이가 무한한 경우

$$
\begin{aligned}
\hat{X}_0 &= \lambda \hat{m}.\hat{X}_2(\hat{X}_1\ \hat{m}) \\
\hat{X}_1 &= \lambda \hat{m}.\hat{m}\{x \mapsto \alpha\{1\}\} \\
\hat{X}_2 &= \lambda \hat{m}.\hat{m} \sqcup (\hat{X}_2\ (\hat{X}_3\ \hat{m})) \\
\hat{X}_3 &= \lambda \hat{m}.\hat{m}\{x \mapsto (\hat{m}\ x) \hat{+} \alpha\{1\}\}
\end{aligned}
$$

$\hat{\mathbb{Z}}$의 높이가 유한한 경우

예를들어, $\hat{\mathbb{Z}} = \{\bot, -, +, \top\}$ 일 때, 위의 방정식은:

$$\begin{aligned}
\hat{X}_0 &= \lambda\hat{m}.\hat{X}_2(\hat{X}_1\,\hat{m}) \\
\hat{X}_1 &= \lambda\hat{m}.\hat{m}\{x \mapsto +\} \\
\hat{X}_2 &= \lambda\hat{m}.\hat{m} \sqcup (\hat{X}_2\,(\hat{X}_3\,\hat{m})) \\
\hat{X}_3 &= \lambda\hat{m}.\hat{m}\{x \mapsto (\hat{m}\,x)\,\hat{+}\,+\}
\end{aligned}$$

프로그램 시작에서의 모든 메모리 상태를 포섭하는 것이

$$\{\} \in \hat{Memory} = Var \xrightarrow{\text{fin}} \hat{Value}$$

라고하면,

$$\hat{X}_0\,\{\}$$

에서 부터 "연쇄반응"을 일으키는 방정식들만 푼다.

연쇄반응을 따라, 풀어야 할 방정식들:

$$\begin{aligned}
\hat{X}_0\,\{\} &= \hat{X}_2(\hat{X}_1\,\{\}) \quad \text{따라서} \\
\hat{X}_1\,\{\} &= \{x \mapsto +\} \quad \text{따라서} \\
\hat{X}_2\,\{x \mapsto +\} &= \{x \mapsto +\} \sqcup (\hat{X}_2\,(\hat{X}_3\,\{x \mapsto +\})) \quad \text{따라서} \\
\hat{X}_3\,\{x \mapsto +\} &= \{x \mapsto +\}\{x \mapsto + \hat{+} +\}
\end{aligned}$$

다시 정리하면

$$\begin{aligned}
\hat{X}_0\,\{\} &= \hat{X}_2\,\{x \mapsto +\} \\
\hat{X}_1\,\{\} &= \{x \mapsto +\} \\
\hat{X}_2\,\{x \mapsto +\} &= \{x \mapsto +\} \sqcup (\hat{X}_2\,\{x \mapsto +\}) \\
\hat{X}_3\,\{x \mapsto +\} &= \{x \mapsto +\}
\end{aligned}$$

위의 방정식을 다시 쓰면 ($\hat{X}_i \star$을 $\hat{Y}_i$로)

$$\begin{aligned}
\hat{Y}_0 &= \hat{Y}_2 \\
\hat{Y}_1 &= \{x \mapsto +\} \\
\hat{Y}_2 &= \{x \mapsto +\} \sqcup \hat{Y}_2 \\
\hat{Y}_3 &= \{x \mapsto +\}
\end{aligned}$$

위의 방정식의 최소해는 고정점 계산(*fixpoint iteration*)
($\sqcup_i (f^i \hat{\bot})$의 계산) 으로:

$$\begin{aligned}
\hat{Y}_0 &= \{x \mapsto +\} \\
\hat{Y}_1 &= \{x \mapsto +\} \\
\hat{Y}_2 &= \{x \mapsto +\} \\
\hat{Y}_3 &= \{x \mapsto +\}
\end{aligned}$$

분석결과: 프로그램의 모든 명령문 실행 후 x는 음이아닌 정수
를 가진다.