Aufgabe 2: Einfache Konstantenpropagation

Optimierende Übersetzer WS 2013

Abgabetermin: 20. November 2013, 16:00 Uhr

1 Einfache intraprozedurale Konstantenpropagation

Die Konstantenpropagation ermittelt für jeden Programmpunkt eine Zuordnung von Programmvariablen zu numerischen Konstanten, sofern am gegebenen Punkt die Variable bei jeder Ausführung des Programms den berechneten konstanten Wert besitzt. Konstante Werte können als Literale im Programm auftreten oder sich durch Anwendung einer Rechenoperation auf konstante Werte ergeben. Die in dieser Aufgabe untersuchte Datenflussanalyse berechnet derartige konstante Werte und reicht die Zuordnung zu Variablen entlang des Kontrollflusses weiter.

Im Folgenden ein Beispiel für einfache intraprozedurale Konstantenpropagation für die Sprache WHILE; nicht zugewiesene Variablen haben dabei Werte, die zum Analysezeitpunkt unbekannt sind:

	ℓ	$\mathrm{CP}_\circ(\ell)$	$\mathrm{CP}_\bullet(\ell)$
$[\texttt{a := 2}]^1\texttt{;}$	1	$\{\}$	$\{\texttt{a} \mapsto 2\}$
$[\texttt{b := 7 + a}]^2\texttt{;}$	2	$\{\texttt{a} \mapsto 2\}$	$\{\texttt{a} \mapsto 2, \texttt{b} \mapsto 9\}$
$\texttt{if } [\texttt{b < c}]^3 \texttt{ then}$	3	$\{\texttt{a} \mapsto 2, \texttt{b} \mapsto 9\}$	$\{\texttt{a} \mapsto 2, \texttt{b} \mapsto 9\}$
$\quad [\texttt{b := b - 3}]^4\texttt{;}$	4	$\{\texttt{a} \mapsto 2, \texttt{b} \mapsto 9\}$	$\{\texttt{a} \mapsto 2, \texttt{b} \mapsto 6\}$
$\texttt{else}$			
$\quad [\texttt{skip}]^5\texttt{;}$	5	$\{\texttt{a} \mapsto 2, \texttt{b} \mapsto 9\}$	$\{\texttt{a} \mapsto 2, \texttt{b} \mapsto 9\}$
$[\texttt{a := a + 7}]^6\texttt{;}$	6	$\{\texttt{a} \mapsto 2\}$	$\{\texttt{a} \mapsto 9\}$

Überlegen Sie sich einige interessante Einschränkungen der Konstantenpropagation:

(a) Die oben beschriebene Analyse errechnet konstante Werte nur für Ausdrücke, deren Operanden ausschließlich konstante Werte sind. Nennen Sie drei Formen von arithmetischen Ausdrücken, deren Ergebnis auch dann als konstanter Wert bestimmt werden kann, wenn ihre Operanden nicht ausschließlich konstante Werte sind.

(b) Beweisen Sie, dass einfache Konstantenpropagation keine distributive Datenflussanalyse ist. Dafür genügt eine Transferfunktion f_ℓ als Gegenbeispiel zur Gleichung $f_\ell(l_1 \sqcup l_2) = f_\ell(l_1) \sqcup f_\ell(l_2)$. Geben Sie daher ein kurzes Beispielprogramm (in WHILE oder C) an, in dem die Analyse für eine Zuweisung aufgrund der Verletzung obiger definierender Eigenschaft keinen konstanten Wert berechnen kann, obwohl der vom Programm berechnete Wert tatsächlich immer derselbe ist.

2 Einfache intraprozedurale Konstantenpropagation mit PAG

Spezifizieren Sie mit PAG eine einfache intraprozedurale Konstantenpropagationsanalyse für die Sprache SL_1, die auf Aufgabenblatt 1 definiert wurde. Ihre Analyse soll also wieder Zuweisungen, `if`-Verzweigungen und `while`-Schleifen behandeln können. Arithmetische Ausdrücke dürfen dabei neben Variablen und Konstanten vom Typ `int` aus den einfachen Rechenoperationen `+`, `-`, `*` und `/` zusammengesetzt sein. Auf der linken Seite jeder Zuweisung steht eine Variable, auf der rechten Seite ein arithmetischer Ausdruck. Logische Ausdrücke in den Bedingungen von Verzweigungen und Schleifen bestehen aus einem der Vergleichsoperatoren `==`, `<` oder `<=` angewendet auf zwei arithmetische Ausdrücke.

Ihre Analyse soll die gesamte Sprache SL_1 mit den oben angeführten arithmetischen Operationen abdecken, Sie dürfen auch zusätzliche arithmetische Operationen behandeln. Für Ausdrücke, die Ihre Analyse nicht behandeln kann, sollte sie informative Warnungen ausgeben; ansonsten ist Ihnen das Verhalten der Analyse auf Programmen, die nicht in der unterstützten Sprache liegen, freigestellt.

Verwenden Sie als Datenstruktur für die Zuordnung zwischen Variablen und Konstanten PAGs 'dynamic functions', die mit dem Typkonstruktor `->` deklariert werden. Für den Wertebereich dieser Abbildung werden Sie einen Typ benötigen, der sowohl konstante Zahlenwerte als auch zumindest einen 'undefinierten' Wert abbilden kann. Ebenso müssen Sie einen Wert für die gänzlich undefinierte Zuordnung an noch nicht analysierten Programmpunkten vorsehen.

3 Abgabe

Senden Sie Ihre Lösungen bis **20. November 2013, 16:00 Uhr** per E-Mail an `gergo@complang.tuwien.ac.at`. Die Betreffzeile der E-Mail soll '`OU: Aufgabe 2,` *Nachname*' lauten. Beantworten Sie Textaufgaben 1 (a) und 1 (b) innerhalb der E-Mail oder hängen Sie Ihre Antworten als Text-, PostScript- oder PDF-Datei an die E-Mail an. Hängen Sie die Analysespezifikation für Teilaufgabe 2 als `.optla`-Datei an die E-Mail an.