

InstCombine: Combine redundant instructions

LLVM optimization

{ Lazar Bećarević 93/2020



Konstrukcija kompilatora 2024/2025

Opis optimizacije

- ⌘ Spajanje instrukcija radi stvaranja manjih, jednostanijih i efikasnijih instrukcija
- ⌘ Obuhvata razne algebarske optimizacije

⌘ Ukoliko se kod binarnog operatora pojavljuje konstanta ona se stavlja na desnu stranu

- ⌘ Arhitekture često imaju "brže" pristupe kada se konstanta koristi kao drugi operand, jer procesor može unapred učitati vrednost konstante i optimizovati instrukciju za bržu obradu.

```
%15 = add nsw i32 5, %14
      ↓
%15 = add nsw i32 %14, 5
```

⌘ Poređenja $<$, $>$, \leq , ili \geq se menjaju sa $=$ ili \neq (gde je to moguće)

- ⌘ Direktnija i jednostavnija operacija

```
%18 = icmp sgt i32 %16, %17
      ↓
%18 = icmp ne i32 %16, %17
```

⌘ Ukoliko se pojavi množenje sa stepenom dvojke menja se šiftovanjem

```
%36 = mul nsw i32 %35, 4096  
      ↓  
%36 = shl i32 %35, 12
```

⌘ Kada se pojavi sabiranje 2 iste vrednosti menjaju se množenjem/šiftovanjem

⌘ Slična logika kao prethodni primer samo što zamena zavisi od ponavljanja promenljivih/konstanti

Zaključak

- ⌘ Od izuzetnog značaja za uređaje sa limitiranom memorijom i limitiranim procesorskim operacijama.
- ⌘ InstCombine omogućava lakšu analizu i dalje optimizacije koda jer generiše jednostavniji i prepoznatljiviji LLVM IR, čineći ga lakšim za dalju obradu od strane drugih optimizacionih rešenja.