Intermediate Representation

Enrico Deiana, Emanuele Del Sozzo

Introduction

clargfwr

IR Instructions

All the instructions are defined by this data structure:

```
type Intruction struct{
        op      int
        arg1    Symbol
        arg2    Symbol
        result  Symbol
        true    *Instruction
        false   *Instruction
        next    *Instruction
}
```

Binary Operations

The binary operations defined in **EL** are:

- arithmetical operations (*SUM*, *SUB*, *MUL*, *DIV*);
- logic operations (*AND*, *OR*);
- comparison operations (*EQUAL*, *NOT_EQUAL*, *LOWER*, *GREATER*, *LOWER_EQUAL*, *GREATER_EQUAL*).

An example is:
EL instruction:

$$x = y + w + z$$

intermediate representation:

$$Istr_1 \quad op : ADD \quad arg1 : y \quad arg2 : w \quad result : x_1 \quad true : NULL \quad false : NULL \quad next : Istr_2$$

$$Istr_2 \quad op : ADD \quad arg1 : x_1 \quad arg2 : z \quad result : x \quad true : NULL \quad false : NULL \quad next : Istr_3$$

$$\dots$$