

JavaScript is disabled on your browser.

- [Prev Class](#)
- [Next Class](#)

- [Frames](#)
- [No Frames](#)

- [All Classes](#)

- [Summary:](#)
- [Nested |](#)
- [Field |](#)
- [Constr |](#)
- [Method](#)

- [Detail:](#)
- [Field |](#)
- [Constr |](#)
- [Method](#)

jminusminus

Class **NLIRInstruction**

- [java.lang.Object](#)
 - [jminusminus.NLIRInstruction](#)
- Direct Known Subclasses:
 - [NLIRALoad](#), [NLIRArithmetic](#), [NLIRAShift](#), [NLIRConditionalJump](#), [NLIRGetField](#), [NLIRGoto](#), [NLIRIntConstant](#), [NLIRInvoke](#), [NLIRLoad](#), [NLIRLoadLocal](#), [NLIRMove](#), [NLIRNewArray](#), [NLIRPhiFunction](#), [NLIRPutField](#), [NLIRReturn](#), [NLIRStore](#), [NLIRStringConstant](#)

<https://powcoder.com>
Add WeChat powcoder

```
abstract class NLIRInstruction  
extends Object
```

Low-level intermediate representation (LIR) of a JVM instruction.

- **Field Summary**

Fields	
Modifier and Type	Field and Description
NBasicBlock	block The block containing this instruction.
int	id Unique identifier of this instruction.
protected static String[]	lirMnemonic Maps JVM opcode to a string mnemonic for LIR instructions.
ArrayList<NRegister>	reads Registers that store the inputs (if any) of this instruction.

NRegister

write

Register that stores the result (if any) of this instruction.

- **Constructor Summary**

Constructors

Modifier

Constructor and Description

protected **NLIRInstruction**(NBasicBlock block, int id)
Construct an NLIRInstruction.

- **Method Summary**

Methods

**Modifier and
Type**

Method and Description

void **allocatePhysicalRegisters()**
Replace references to virtual registers in this LIR instruction with references to physical registers.

void **toSpim**(PrintWriter out)
Translate this LIR instruction into SPIM and write it out to the specified output stream.

String **toString()**
Return a string representation of this instruction.

- **Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

- **Field Detail**

- **lirMnemonic**

protected static String[] lirMnemonic

Maps JVM opcode to a string mnemonic for LIR instructions. For example, imul is mapped to the "MUL".

- **block**

public NBasicBlock block

The block containing this instruction.

- **id**

public int id

Unique identifier of this instruction.

- **reads**

public ArrayList<NRegister> reads

Registers that store the inputs (if any) of this instruction.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

- **write**

public **NRegister** write

Register that stores the result (if any) of this instruction.

- **Constructor Detail**

- **NLIRInstruction**

protected **NLIRInstruction**(**NBasicBlock** block,
int id)

Construct an **NLIRInstruction**.

Parameters:

block - enclosing block.
id - identifier of the instruction.

- **Method Detail**

- **allocatePhysicalRegisters**

public void allocatePhysicalRegisters()

Replace references to virtual registers in this LIR instruction with references to physical registers.

- **toSpim**

public void toSpim(**PrintWriter** out)

Translate this LIR instruction into SPIM and write it out to the specified output stream.

Parameters:

out - output stream for SPIM code

- **toString**

public **String** toString()

Return a string representation of this instruction.

Overrides:

toString in class **Object**

Returns:

string representation of this instruction.

- Overview
- Package
- **Class**
- Tree
- Deprecated
- Index
- Help

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

- **Prev Class**
- **Next Class**

- Frames
- No Frames

- All Classes

- Summary:
- Nested |
- Field |
- Constr |
- Method

- Detail:
- Field |
- Constr |
- Method

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder