

## NRegister.java

```
1  // Copyright 2013 Bill Campbell, Swami Iyer and Bahar Akbal-Delibas
2
3  package jminusminus;
4
5  import java.util.ArrayList;
6  import java.util.Collections;
7
8  /**
9   * An abstraction for a (virtual or physical) register.
10  */
11
12  abstract class NRegister {
13
14      /** Register number. */
15      protected int number;
16
17      /** Register name. */
18      protected String name;
19
20      /**
21       * Construct an NRegister.
22       *
23       * @param number
24       *           register number.
25       * @param name
26       *           register name.
27       */
28
29      protected NRegister(int number, String name) {
30          this.number = number;
31          this.name = name;
32      }
33
34      /**
35       * Return the number of this register.
36       *
37       * @return register number.
38       */
39
40      public int number() {
41          return number;
42      }
43
44      /**
45       * Return the name of this register.
46       *
47       * @return register name.
48       */
49
50      public String name() {
51          return name;
52      }
53
54  }
55
56  /**
57   * An abstraction for a virtual register.
58   */
59
60  class NVirtualRegister extends NRegister {
61
62      /** Type (short name) of value in register. */
63      private String sType;
64
65      /** Type (long name) of value in register. */
66      private String lType;
```

```

67
68 /**
69  * Construct an NVirtualRegister.
70  *
71  * @param number
72  *         register number.
73  * @param sType
74  *         type (short name) of value in register.
75  * @param lType
76  *         type (long name) of value in register.
77  */
78
79 public NVirtualRegister(int number, String sType, String lType) {
80     super(number, "V" + number);
81     this.sType = sType;
82     this.lType = lType;
83 }
84
85 /**
86  * Return a string representation of this virtual register.
87  *
88  * @return string representation.
89  */
90
91 public String toString() {
92     return "[" + name + "|" + sType + "]";
93 }
94
95 }
96
97 /**
98  * An abstraction for a physical (SPIM) register.
99  */
100
101 class NPhysicalRegister extends NRegister {
102
103     /**
104      * Maximum number of physical registers used for allocation, starting at T0.
105      */
106     public static int MAX_COUNT = 8;
107
108     // Constants identifying the physical registers. These
109     // can be used as indices into the static regInfo array
110     // to access the representations of the corresponding
111     // registers.
112
113     /** Constant 0. */
114     public static final int ZERO = 0;
115
116     /** Reserved for assembler. */
117     public static final int AT = 1;
118
119     /** Expression evaluation and results of a function. */
120     public static final int V0 = 2;
121
122     /** Expression evaluation and results of a function. */
123     public static final int V1 = 3;
124
125     /** Argument 1. */
126     public static final int A0 = 4;
127
128     /** Argument 2. */
129     public static final int A1 = 5;
130
131     /** Argument 3. */
132     public static final int A2 = 6;
133
134     /** Argument 4. */
135     public static final int A3 = 7;

```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

```

136  /** Temporary (not preserved across call). */
137  public static final int T0 = 8;
138
139  /** Temporary (not preserved across call). */
140  public static final int T1 = 9;
141
142  /** Temporary (not preserved across call). */
143  public static final int T2 = 10;
144
145  /** Temporary (not preserved across call). */
146  public static final int T3 = 11;
147
148  /** Temporary (not preserved across call). */
149  public static final int T4 = 12;
150
151  /** Temporary (not preserved across call). */
152  public static final int T5 = 13;
153
154  /** Temporary (not preserved across call). */
155  public static final int T6 = 14;
156
157  /** Temporary (not preserved across call). */
158  public static final int T7 = 15;
159
160  /** Temporary (preserved across call). */
161  public static final int S0 = 16;
162
163  /** Temporary (preserved across call). */
164  public static final int S1 = 17;
165
166  /** Temporary (preserved across call). */
167  public static final int S2 = 18;
168
169  /** Temporary (preserved across call). */
170  public static final int S3 = 19;
171
172  /** Temporary (preserved across call). */
173  public static final int S4 = 20;
174
175  /** Temporary (preserved across call). */
176  public static final int S5 = 21;
177
178  /** Temporary (preserved across call). */
179  public static final int S6 = 22;
180
181  /** Temporary (preserved across call). */
182  public static final int S7 = 23;
183
184  /** Temporary (not preserved across call). */
185  public static final int T8 = 24;
186
187  /** Temporary (not preserved across call). */
188  public static final int T9 = 25;
189
190  /** Reserved for OS kernel. */
191  public static final int K0 = 26;
192
193  /** Reserved for OS kernel. */
194  public static final int K1 = 27;
195
196  /** Pointer to global area. */
197  public static final int GP = 28;
198
199  /** Stack pointer. */
200  public static final int SP = 29;
201
202  /** Frame pointer. */
203  public static final int FP = 30;
204

```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

```

205
206 /** Return address (used by function call). */
207 public static final int RA = 31;
208
209 /**
210  * Maps register number to the register's representation.
211  */
212 public static final NPhysicalRegister[] regInfo = {
213     new NPhysicalRegister(0, "zero"), new NPhysicalRegister(1, "at"),
214     new NPhysicalRegister(2, "v0"), new NPhysicalRegister(3, "v1"),
215     new NPhysicalRegister(4, "a0"), new NPhysicalRegister(5, "a1"),
216     new NPhysicalRegister(6, "a2"), new NPhysicalRegister(7, "a3"),
217     new NPhysicalRegister(8, "t0"), new NPhysicalRegister(9, "t1"),
218     new NPhysicalRegister(10, "t2"), new NPhysicalRegister(11, "t3"),
219     new NPhysicalRegister(12, "t4"), new NPhysicalRegister(13, "t5"),
220     new NPhysicalRegister(14, "t6"), new NPhysicalRegister(15, "t7"),
221     new NPhysicalRegister(16, "s0"), new NPhysicalRegister(17, "s1"),
222     new NPhysicalRegister(18, "s2"), new NPhysicalRegister(19, "s3"),
223     new NPhysicalRegister(20, "s4"), new NPhysicalRegister(21, "s5"),
224     new NPhysicalRegister(22, "s6"), new NPhysicalRegister(23, "s7"),
225     new NPhysicalRegister(24, "t8"), new NPhysicalRegister(25, "t9"),
226     new NPhysicalRegister(26, "k0"), new NPhysicalRegister(27, "k1"),
227     new NPhysicalRegister(28, "gp"), new NPhysicalRegister(29, "sp"),
228     new NPhysicalRegister(30, "fp"), new NPhysicalRegister(31, "ra"), };
229
230 /**
231  * Construct an NPhysicalRegister.
232  *
233  * @param number number of the register.
234  * @param name name of the register.
235  */
236
237
238
239 public NPhysicalRegister(int number, String name) {
240     super(number, name);
241 }
242
243 /**
244  * Return a string representation of this physical register.
245  *
246  * @return string representation.
247  */
248
249 public String toString() {
250     return "$" + name();
251 }
252
253 }
254

```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder