```java
// Copyright 2013 Bill Campbell, Swami Iyer and Bahar Akbal-Delibas

package jminusminus;

/**
 * Constants used within CL*.java files.
 */

public class CLConstants {

    /**
     * Magic number (0xCAFEBABE) identifying the class file format.
     */
    public static final long MAGIC = 3405691582L;

    /** Major version for the class files that j-- compiles. */
    public static final int MAJOR_VERSION = 49;

    /** Minor version for the class files that j-- compiles. */
    public static final int MINOR_VERSION = 0;

    /** public access flag. */
    public static final int ACC_PUBLIC = 0x0001;

    /** private access flag. */
    public static final int ACC_PRIVATE = 0x0002;

    /** protected access flag. */
    public static final int ACC_PROTECTED = 0x0004;

    /** static access flag. */
    public static final int ACC_STATIC = 0x0008;

    /** final access flag. */
    public static final int ACC_FINAL = 0x0010;

    /** super access flag. */
    public static final int ACC_SUPER = 0x0020;

    /** synchronized access flag. */
    public static final int ACC_SYNCHRONIZED = 0x0020;

    /** volatile access flag. */
    public static final int ACC_VOLATILE = 0x0040;

    /** bridge access flag. */
    public static final int ACC_BRIDGE = 0x0040;

    /** transient access flag. */
    public static final int ACC_TRANSIENT = 0x0080;

    /** varargs access flag. */
    public static final int ACC_VARARGS = 0x0080;

    /** native access flag. */
    public static final int ACC_NATIVE = 0x0100;

    /** interface access flag. */
    public static final int ACC_INTERFACE = 0x0200;

    /** abstract access flag. */
    public static final int ACC_ABSTRACT = 0x0400;

    /** strict access flag. */
    public static final int ACC_STRICT = 0x0800;

```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
67      /** synthetic access flag. */
68      public static final int ACC_SYNTHETIC = 0x1000;
69
70      /** annotation access flag. */
71      public static final int ACC_ANNOTATION = 0x2000;
72
73      /** enum access flag. */
74      public static final int ACC_ENUM = 0x4000;
75
76      /** Identifies CONSTANT_Utf8_info constant pool structure. */
77      public static final short CONSTANT_Utf8 = 1;
78
79      /** Identifies CONSTANT_Integer_info constant pool structure. */
80      public static final short CONSTANT_Integer = 3;
81
82      /** Identifies CONSTANT_Float_info constant pool structure. */
83      public static final short CONSTANT_Float = 4;
84
85      /** Identifies CONSTANT_Long_info constant pool structure. */
86      public static final short CONSTANT_Long = 5;
87
88      /** Identifies CONSTANT_Double_info constant pool structure. */
89      public static final short CONSTANT_Double = 6;
90
91      /** Identifies CONSTANT_Class_info constant pool structure. */
92      public static final short CONSTANT_Class = 7;
93
94      /** Identifies CONSTANT_String_info constant pool structure. */
95      public static final short CONSTANT_String = 8;
96
97      /** Identifies CONSTANT_Fieldref_info constant pool structure. */
98      public static final short CONSTANT_Fieldref = 9;
99
100     /**
101      * Identifies CONSTANT_Methodref_info constant pool structure.
102      */
103     public static final short CONSTANT_Methodref = 10;
104
105     /**
106      * Identifies CONSTANT_InterfaceMethodref_info constant pool structure.
107      */
108     public static final short CONSTANT_InterfaceMethodref = 11;
109
110     /**
111      * Identifies CONSTANT_NameAndType_info constant pool structure.
112      */
113     public static final short CONSTANT_NameAndType = 12;
114
115     /** Identifies ConstantValue attribute. */
116     public static final String ATT_CONSTANT_VALUE = "ConstantValue";
117
118     /** Identifies Code attribute. */
119     public static final String ATT_CODE = "Code";
120
121     /** Identifies Exceptions attribute. */
122     public static final String ATT_EXCEPTIONS = "Exceptions";
123
124     /** Identifies InnerClasses attribute. */
125     public static final String ATT_INNER_CLASSES = "InnerClasses";
126
127     /** Identifies EnclosingMethod attribute. */
128     public static final String ATT_ENCLOSING_METHOD = "EnclosingMethod";
129
130     /** Identifies Synthetic attribute. */
131     public static final String ATT_SYNTHETIC = "Synthetic";
132
133     /** Identifies Signature attribute. */
134     public static final String ATT_SIGNATURE = "Signature";
135
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
136     /** Identifies SourceFile attribute. */
137     public static final String ATT_SOURCE_FILE = "SourceFile";
138
139     /** Identifies SourceDebugExtension attribute. */
140     public static final String ATT_SOURCE_DEBUG_EXTENSION =
"SourceDebugExtension";
141
142     /** Identifies LineNumberTable attribute. */
143     public static final String ATT_LINE_NUMBER_TABLE = "LineNumberTable";
144
145     /** Identifies LocalVariableTable attribute. */
146     public static final String ATT_LOCAL_VARIABLE_TABLE = "LocalVariableTable";
147
148     /** Identifies LocalVariableTypeTable attribute. */
149     public static final String ATT_LOCAL_VARIABLE_TYPE_TABLE =
"LocalVariableTypeTable";
150
151     /** Identifies Deprecated attribute. */
152     public static final String ATT_DEPRECATED = "Deprecated";
153
154     /** Identifies RuntimeVisibleAnnotations attribute. */
155     public static final String ATT_RUNTIME_VISIBLE_ANNOTATIONS =
"RuntimeVisibleAnnotations";
156
157     /** Identifies RuntimeInvisibleAnnotations attribute. */
158     public static final String ATT_RUNTIME_INVISIBLE_ANNOTATIONS =
"RuntimeInvisibleAnnotations";
159
160     /** Identifies RuntimeVisibleParameterAnnotations attribute. */
161     public static final String ATT_RUNTIME_VISIBLE_PARAMETER_ANNOTATIONS =
"RuntimeVisibleParameterAnnotations";
162
163     /** Identifies RuntimeInvisibleParameterAnnotations attribute. */
164     public static final String ATT_RUNTIME_INVISIBLE_PARAMETER_ANNOTATIONS =
"RuntimeInvisibleParameterAnnotations";
165
166     /** Identifies AnnotationDefault attribute. */
167     public static final String ATT_ANNOTATION_DEFAULT = "AnnotationDefault";
168
169     /** Identifies boolean type of annotation element value. */
170     public static final short ELT_B = 'B';
171
172     /** Identifies char type of annotation element value. */
173     public static final short ELT_C = 'C';
174
175     /** Identifies double type of annotation element value. */
176     public static final short ELT_D = 'D';
177
178     /** Identifies float type of annotation element value. */
179     public static final short ELT_F = 'F';
180
181     /** Identifies int type of annotation element value. */
182     public static final short ELT_I = 'I';
183
184     /** Identifies long type of annotation element value. */
185     public static final short ELT_J = 'J';
186
187     /** Identifies short type of annotation element value. */
188     public static final short ELT_S = 'S';
189
190     /** Identifies boolean type of annotation element value. */
191     public static final short ELT_Z = 'Z';
192
193     /** Identifies String type of annotation element value. */
194     public static final short ELT_s = 's';
195
196     /** Identifies class type of annotation element value. */
197     public static final short ELT_c = 'c';
198
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
199     /** Identifies annotation type of annotation element value. */
200     public static final short ELT_ANNOTATION = '@';
201
202     /** Identifies array type of annotation element value. */
203     public static final short ELT_ARRAY = '[';
204
205     /** Identifies enum type of annotation element value. */
206     public static final short ELT_e = 'e';
207
208     // JVM instructions begin here
209
210     /** NOP instruction. */
211     public static final int NOP = 0;
212
213     /** ACONST_NULL instruction. */
214     public static final int ACONST_NULL = 1;
215
216     /** ICONST_M1 instruction. */
217     public static final int ICONST_M1 = 2;
218
219     /** ICONST_0 instruction. */
220     public static final int ICONST_0 = 3;
221
222     /** ICONST_1 instruction. */
223     public static final int ICONST_1 = 4;
224
225     /** ICONST_2 instruction. */
226     public static final int ICONST_2 = 5;
227
228     /** ICONST_3 instruction. */
229     public static final int ICONST_3 = 6;
230
231     /** ICONST_4 instruction. */
232     public static final int ICONST_4 = 7;
233
234     /** ICONST_5 instruction. */
235     public static final int ICONST_5 = 8;
236
237     /** LCONST_0 instruction. */
238     public static final int LCONST_0 = 9;
239
240     /** LCONST_1 instruction. */
241     public static final int LCONST_1 = 10;
242
243     /** FCONST_0 instruction. */
244     public static final int FCONST_0 = 11;
245
246     /** FCONST_1 instruction. */
247     public static final int FCONST_1 = 12;
248
249     /** FCONST_2 instruction. */
250     public static final int FCONST_2 = 13;
251
252     /** DCONST_0 instruction. */
253     public static final int DCONST_0 = 14;
254
255     /** DCONST_1 instruction. */
256     public static final int DCONST_1 = 15;
257
258     /** BIPUSH instruction. */
259     public static final int BIPUSH = 16;
260
261     /** SIPUSH instruction. */
262     public static final int SIPUSH = 17;
263
264     /** LDC instruction. */
265     public static final int LDC = 18;
266
267     /** LDC_W instruction. */
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
268     public static final int LDC_W = 19;
269
270     /** LDC2_W instruction. */
271     public static final int LDC2_W = 20;
272
273     /** ILOAD instruction. */
274     public static final int ILOAD = 21;
275
276     /** LLOAD instruction. */
277     public static final int LLOAD = 22;
278
279     /** FLOAD instruction. */
280     public static final int FLOAD = 23;
281
282     /** DLOAD instruction. */
283     public static final int DLOAD = 24;
284
285     /** ALOAD instruction. */
286     public static final int ALOAD = 25;
287
288     /** ILOAD_0 instruction. */
289     public static final int ILOAD_0 = 26;
290
291     /** ILOAD_1 instruction. */
292     public static final int ILOAD_1 = 27;
293
294     /** ILOAD_2 instruction. */
295     public static final int ILOAD_2 = 28;
296
297     /** ILOAD_3 instruction. */
298     public static final int ILOAD_3 = 29;
299
300     /** LLOAD_0 instruction. */
301     public static final int LLOAD_0 = 30;
302
303     /** LLOAD_1 instruction. */
304     public static final int LLOAD_1 = 31;
305
306     /** LLOAD_2 instruction. */
307     public static final int LLOAD_2 = 32;
308
309     /** LLOAD_3 instruction. */
310     public static final int LLOAD_3 = 33;
311
312     /** FLOAD_0 instruction. */
313     public static final int FLOAD_0 = 34;
314
315     /** FLOAD_1 instruction. */
316     public static final int FLOAD_1 = 35;
317
318     /** FLOAD_2 instruction. */
319     public static final int FLOAD_2 = 36;
320
321     /** FLOAD_3 instruction. */
322     public static final int FLOAD_3 = 37;
323
324     /** DLOAD_0 instruction. */
325     public static final int DLOAD_0 = 38;
326
327     /** DLOAD_1 instruction. */
328     public static final int DLOAD_1 = 39;
329
330     /** DLOAD_2 instruction. */
331     public static final int DLOAD_2 = 40;
332
333     /** DLOAD_3 instruction. */
334     public static final int DLOAD_3 = 41;
335
336     /** ALOAD_0 instruction. */
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
337     public static final int ALOAD_0 = 42;
338
339     /** ALOAD_1 instruction. */
340     public static final int ALOAD_1 = 43;
341
342     /** ALOAD_2 instruction. */
343     public static final int ALOAD_2 = 44;
344
345     /** ALOAD_3 instruction. */
346     public static final int ALOAD_3 = 45;
347
348     /** IALOAD instruction. */
349     public static final int IALOAD = 46;
350
351     /** LALOAD instruction. */
352     public static final int LALOAD = 47;
353
354     /** FALOAD instruction. */
355     public static final int FALOAD = 48;
356
357     /** DALOAD instruction. */
358     public static final int DALOAD = 49;
359
360     /** AALOAD instruction. */
361     public static final int AALOAD = 50;
362
363     /** BALOAD instruction. */
364     public static final int BALOAD = 51;
365
366     /** CALOAD instruction. */
367     public static final int CALOAD = 52;
368
369     /** SALOAD instruction. */
370     public static final int SALOAD = 53;
371
372     /** ISTORE instruction. */
373     public static final int ISTORE = 54;
374
375     /** LSTORE instruction. */
376     public static final int LSTORE = 55;
377
378     /** FSTORE instruction. */
379     public static final int FSTORE = 56;
380
381     /** DSTORE instruction. */
382     public static final int DSTORE = 57;
383
384     /** ASTORE instruction. */
385     public static final int ASTORE = 58;
386
387     /** ISTORE_0 instruction. */
388     public static final int ISTORE_0 = 59;
389
390     /** ISTORE_1 instruction. */
391     public static final int ISTORE_1 = 60;
392
393     /** ISTORE_2 instruction. */
394     public static final int ISTORE_2 = 61;
395
396     /** ISTORE_3 instruction. */
397     public static final int ISTORE_3 = 62;
398
399     /** LSTORE_0 instruction. */
400     public static final int LSTORE_0 = 63;
401
402     /** LSTORE_1 instruction. */
403     public static final int LSTORE_1 = 64;
404
405     /** LSTORE_2 instruction. */
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
406     public static final int LSTORE_2 = 65;
407
408     /** LSTORE_3 instruction. */
409     public static final int LSTORE_3 = 66;
410
411     /** FSTORE_0 instruction. */
412     public static final int FSTORE_0 = 67;
413
414     /** FSTORE_1 instruction. */
415     public static final int FSTORE_1 = 68;
416
417     /** FSTORE_2 instruction. */
418     public static final int FSTORE_2 = 69;
419
420     /** FSTORE_3 instruction. */
421     public static final int FSTORE_3 = 70;
422
423     /** DSTORE_0 instruction. */
424     public static final int DSTORE_0 = 71;
425
426     /** DSTORE_1 instruction. */
427     public static final int DSTORE_1 = 72;
428
429     /** DSTORE_2 instruction. */
430     public static final int DSTORE_2 = 73;
431
432     /** DSTORE_3 instruction. */
433     public static final int DSTORE_3 = 74;
434
435     /** ASTORE_0 instruction. */
436     public static final int ASTORE_0 = 75;
437
438     /** ASTORE_1 instruction. */
439     public static final int ASTORE_1 = 76;
440
441     /** ASTORE_2 instruction. */
442     public static final int ASTORE_2 = 77;
443
444     /** ASTORE_3 instruction. */
445     public static final int ASTORE_3 = 78;
446
447     /** IASTORE instruction. */
448     public static final int IASTORE = 79;
449
450     /** LASTORE instruction. */
451     public static final int LASTORE = 80;
452
453     /** FASTORE instruction. */
454     public static final int FASTORE = 81;
455
456     /** DASTORE instruction. */
457     public static final int DASTORE = 82;
458
459     /** AASTORE instruction. */
460     public static final int AASTORE = 83;
461
462     /** BASTORE instruction. */
463     public static final int BASTORE = 84;
464
465     /** CASTORE instruction. */
466     public static final int CASTORE = 85;
467
468     /** SASTORE instruction. */
469     public static final int SASTORE = 86;
470
471     /** POP instruction. */
472     public static final int POP = 87;
473
474     /** POP2 instruction. */
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
475     public static final int POP2 = 88;
476
477     /** DUP instruction. */
478     public static final int DUP = 89;
479
480     /** DUP_X1 instruction. */
481     public static final int DUP_X1 = 90;
482
483     /** DUP_X2 instruction. */
484     public static final int DUP_X2 = 91;
485
486     /** DUP2 instruction. */
487     public static final int DUP2 = 92;
488
489     /** DUP2_X1 instruction. */
490     public static final int DUP2_X1 = 93;
491
492     /** DUP2_X2 instruction. */
493     public static final int DUP2_X2 = 94;
494
495     /** SWAP instruction. */
496     public static final int SWAP = 95;
497
498     /** IADD instruction. */
499     public static final int IADD = 96;
500
501     /** LADD instruction. */
502     public static final int LADD = 97;
503
504     /** FADD instruction. */
505     public static final int FADD = 98;
506
507     /** DADD instruction. */
508     public static final int DADD = 99;
509
510     /** ISUB instruction. */
511     public static final int ISUB = 100;
512
513     /** LSUB instruction. */
514     public static final int LSUB = 101;
515
516     /** FSUB instruction. */
517     public static final int FSUB = 102;
518
519     /** DSUB instruction. */
520     public static final int DSUB = 103;
521
522     /** IMUL instruction. */
523     public static final int IMUL = 104;
524
525     /** LMUL instruction. */
526     public static final int LMUL = 105;
527
528     /** FMUL instruction. */
529     public static final int FMUL = 106;
530
531     /** DMUL instruction. */
532     public static final int DMUL = 107;
533
534     /** IDIV instruction. */
535     public static final int IDIV = 108;
536
537     /** LDIV instruction. */
538     public static final int LDIV = 109;
539
540     /** FDIV instruction. */
541     public static final int FDIV = 110;
542
543     /** DDIV instruction. */
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
544     public static final int DDIV = 111;
545
546     /** IREM instruction. */
547     public static final int IREM = 112;
548
549     /** LREM instruction. */
550     public static final int LREM = 113;
551
552     /** FREM instruction. */
553     public static final int FREM = 114;
554
555     /** DREM instruction. */
556     public static final int DREM = 115;
557
558     /** INEG instruction. */
559     public static final int INEG = 116;
560
561     /** LNEG instruction. */
562     public static final int LNEG = 117;
563
564     /** FNEG instruction. */
565     public static final int FNEG = 118;
566
567     /** DNEG instruction. */
568     public static final int DNEG = 119;
569
570     /** ISHL instruction. */
571     public static final int ISHL = 120;
572
573     /** LSHL instruction. */
574     public static final int LSHL = 121;
575
576     /** ISHR instruction. */
577     public static final int ISHR = 122;
578
579     /** LSHR instruction. */
580     public static final int LSHR = 123;
581
582     /** IUSHR instruction. */
583     public static final int IUSHR = 124;
584
585     /** LUSHR instruction. */
586     public static final int LUSHR = 125;
587
588     /** IAND instruction. */
589     public static final int IAND = 126;
590
591     /** LAND instruction. */
592     public static final int LAND = 127;
593
594     /** IOR instruction. */
595     public static final int IOR = 128;
596
597     /** LOR instruction. */
598     public static final int LOR = 129;
599
600     /** IXOR instruction. */
601     public static final int IXOR = 130;
602
603     /** LXOR instruction. */
604     public static final int LXOR = 131;
605
606     /** IINC instruction. */
607     public static final int IINC = 132;
608
609     /** I2L instruction. */
610     public static final int I2L = 133;
611
612     /** I2F instruction. */
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
613     public static final int I2F = 134;
614
615     /** I2D instruction. */
616     public static final int I2D = 135;
617
618     /** L2I instruction. */
619     public static final int L2I = 136;
620
621     /** L2F instruction. */
622     public static final int L2F = 137;
623
624     /** L2D instruction. */
625     public static final int L2D = 138;
626
627     /** F2I instruction. */
628     public static final int F2I = 139;
629
630     /** F2L instruction. */
631     public static final int F2L = 140;
632
633     /** F2D instruction. */
634     public static final int F2D = 141;
635
636     /** D2I instruction. */
637     public static final int D2I = 142;
638
639     /** D2L instruction. */
640     public static final int D2L = 143;
641
642     /** D2F instruction. */
643     public static final int D2F = 144;
644
645     /** I2B instruction. */
646     public static final int I2B = 145;
647
648     /** I2C instruction. */
649     public static final int I2C = 146;
650
651     /** I2S instruction. */
652     public static final int I2S = 147;
653
654     /** LCMP instruction. */
655     public static final int LCMP = 148;
656
657     /** FCMPL instruction. */
658     public static final int FCMPL = 149;
659
660     /** FCMPG instruction. */
661     public static final int FCMPG = 150;
662
663     /** DCMPL instruction. */
664     public static final int DCMPL = 151;
665
666     /** DCMPG instruction. */
667     public static final int DCMPG = 152;
668
669     /** IFEQ instruction. */
670     public static final int IFEQ = 153;
671
672     /** IFNE instruction. */
673     public static final int IFNE = 154;
674
675     /** IFLT instruction. */
676     public static final int IFLT = 155;
677
678     /** IFGE instruction. */
679     public static final int IFGE = 156;
680
681     /** IFGT instruction. */
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
682     public static final int IFGT = 157;
683
684     /** IFLE instruction. */
685     public static final int IFLE = 158;
686
687     /** IF_ICMPEQ instruction. */
688     public static final int IF_ICMPEQ = 159;
689
690     /** IF_ICMPNE instruction. */
691     public static final int IF_ICMPNE = 160;
692
693     /** IF_ICMPLT instruction. */
694     public static final int IF_ICMPLT = 161;
695
696     /** IF_ICMPGE instruction. */
697     public static final int IF_ICMPGE = 162;
698
699     /** IF_ICMPGT instruction. */
700     public static final int IF_ICMPGT = 163;
701
702     /** IF_ICMPLE instruction. */
703     public static final int IF_ICMPLE = 164;
704
705     /** IF_ACMPEQ instruction. */
706     public static final int IF_ACMPEQ = 165;
707
708     /** IF_ACMPNE instruction. */
709     public static final int IF_ACMPNE = 166;
710
711     /** GOTO instruction. */
712     public static final int GOTO = 167;
713
714     /** JSR instruction. */
715     public static final int JSR = 168;
716
717     /** RET instruction. */
718     public static final int RET = 169;
719
720     /** TABLESWITCH instruction. */
721     public static final int TABLESWITCH = 170;
722
723     /** LOOKUPSWITCH instruction. */
724     public static final int LOOKUPSWITCH = 171;
725
726     /** IRETURN instruction. */
727     public static final int IRETURN = 172;
728
729     /** LRETURN instruction. */
730     public static final int LRETURN = 173;
731
732     /** FRETURN instruction. */
733     public static final int FRETURN = 174;
734
735     /** DRETURN instruction. */
736     public static final int DRETURN = 175;
737
738     /** ARETURN instruction. */
739     public static final int ARETURN = 176;
740
741     /** RETURN instruction. */
742     public static final int RETURN = 177;
743
744     /** GETSTATIC instruction. */
745     public static final int GETSTATIC = 178;
746
747     /** PUTSTATIC instruction. */
748     public static final int PUTSTATIC = 179;
749
750     /** GETFIELD instruction. */
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
    public static final int GETFIELD = 180;

    /** PUTFIELD instruction. */
    public static final int PUTFIELD = 181;

    /** INVOKEVIRTUAL instruction. */
    public static final int INVOKEVIRTUAL = 182;

    /** INVOKESPECIAL instruction. */
    public static final int INVOKESPECIAL = 183;

    /** INVOKESTATIC instruction. */
    public static final int INVOKESTATIC = 184;

    /** INVOKEINTERFACE instruction. */
    public static final int INVOKEINTERFACE = 185;

    /** INVOKEDYNAMIC instruction. */
    public static final int INVOKEDYNAMIC = 186;

    /** NEW instruction. */
    public static final int NEW = 187;

    /** NEWARRAY instruction. */
    public static final int NEWARRAY = 188;

    /** ANEWARRAY instruction. */
    public static final int ANEWARRAY = 189;

    /** ARRAYLENGTH instruction. */
    public static final int ARRAYLENGTH = 190;

    /** ATHROW instruction. */
    public static final int ATHROW = 191;

    /** CHECKCAST instruction. */
    public static final int CHECKCAST = 192;

    /** INSTANCEOF instruction. */
    public static final int INSTANCEOF = 193;

    /** MONITORENTER instruction. */
    public static final int MONITORENTER = 194;

    /** MONITOREXIT instruction. */
    public static final int MONITOREXIT = 195;

    /** WIDE instruction. */
    public static final int WIDE = 196;

    /** MULTIANEWARRAY instruction. */
    public static final int MULTIANEWARRAY = 197;

    /** IFNULL instruction. */
    public static final int IFNULL = 198;

    /** IFNONNULL instruction. */
    public static final int IFNONNULL = 199;

    /** GOTO_W instruction. */
    public static final int GOTO_W = 200;

    /** JSR_W instruction. */
    public static final int JSR_W = 201;

    // JVM instructions end here

    /**
     * We classify the JVM instructions into the following categories.
```

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

```
820      */
821     enum Category {
822         OBJECT, FIELD, METHOD1, METHOD2, ARRAY1, ARRAY2, ARRAY3, ARITHMETIC1,
ARITHMETIC2, BIT, COMPARISON, CONVERSION, FLOW_CONTROL1, FLOW_CONTROL2,
FLOW_CONTROL3, FLOW_CONTROL4, LOAD_STORE1, LOAD_STORE2, LOAD_STORE3, LOAD_STORE4,
STACK, MISC;
823     }
824
825     // The constants below simply serve as markers. We are not
826     // interested in their values, which however have been picked
827     // so
828     // as not to conflict with others.
829
830     /**
831      * Denotes values that are irrelevant to certain instructions. For example,
832      * local variable index for arithmetic instructions.
833      */
834     public static final int IRRELEVANT = -1;
835
836     /**
837      * Denotes values that are not statically known. For example, stack units
838      * for field instructions.
839      */
840     public static final int DYNAMIC = 300;
841
842     /**
843      * Stack units for the instructions that empty the operand stack.
844      */
845     public static final int EMPTY_STACK = 301;
846
847     /**
848      * Stack units for the instructions that set the operand stack to unit size.
849      */
850     public static final int UNIT_SIZE_STACK = 302;
851
852 }
853
```

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder