COMP3131/9102: Programming Languages and Compilers

Week 10 Tutorial Solutions

Jasmin Assembly Language

1.

;; Produced by JasminVisitor (BCEL)
;; http://www.inf.fu-berlin.de/~dahm/BCEL/
;; Mon Oct 01 14:38:06 EST 2001

.source Test.java
.class Test
.super java/lang/Object

.method <init>()V
.limit stack 1
.limit locals 1
.var 0 is this LTest; from Label0 to Label1

Label0:
.line 1
   aload_0
   invokespecial java/lang/Object/<init>()V

Label1:
   return

.end method

.method fi(II)I
.limit stack 2
.limit locals 3
.var 0 is this LTest; from Label1 to Label2
.var 1 is arg0 I from Label1 to Label2
.var 2 is arg1 I from Label1 to Label2

Label1:
.line 4
   iload_1
   iload_2
   if_icmple Label0

.line 5
   iconst_1

1
ireturn
Label0:
  .line 7
  icont_2
Label2:
  ireturn
.end method

.method  ff(FZ)Z
  .limit stack 4
  .limit locals 3
  .var 0 is this LTest; from Label4 to Label3
  .var 1 is arg0 F from Label4 to Label3
  .var 2 is arg1 Z from Label4 to Label3
Label4:
  .line 11
     fload_1
     f2d
     dconst_1
     dcml
     il0t Label0
     iload_2
     ifeq Label0
Label0:
  .line 14
     iload_2
     ireturn
Label2:
  .line 12
     iload_2
     ireturn
Label3:
.end method

.method public static main([Ljava/lang/String;)V
  .limit stack 4
  .limit locals 2
  .var 0 is arg0 [Ljava/lang/String; from Label0 to Label1

2
Label0:
  .line 19
  new Test
dup
  invokespecial Test/<init>()V
  astore_1
  .line 20
  getstatic java.lang.System.out Ljava/io/PrintStream;
  aload_1
  iconst_1
  iconst_2
  invokevirtual Test/fi(II)I
  invokevirtual java/io/PrintStream/println(I)V
  .line 21
  getstatic java.lang.System.out Ljava/io/PrintStream;
  aload_1
  fconst_2
  iconst_1
  invokevirtual Test/ff(FZ)Z
  invokevirtual java/io/PrintStream/println(Z)V

Label1:
  .line 22
  return

  .end method

2.

To keep the instruction set small, not all operations are supported for all types. Most operations on the types boolean, byte, char and short are not supported by the JVM. These types are converted to int, operated on as int and finally converted to respective types before being stored.

If i2b were removed, the output would be:

0 1 2 ... 2147483647 -2147483648 -2147483647 .. -1 0 1 2 ...

where 2147483648 = 2^{31}. 

3