

```

20 import comp
21 import firm.BackEdges;
22 import firm.BackWalker;
23 import firm.BlockWalker;
24 import firm.Graph;
25 import firm.Program;
26 import firm.nodes.Block;
27
28 public final class AssemblerGenerator {
29     private AssemblerGenerator() {
30     }
31
32     public static void createAssemblerX8664(Path outputFile, final CallingConvention callingConvention,
33         boolean debugRegisterAllocation) throws IOException {
34         InterferenceGraph.setDebuggingMode(debugRegisterAllocation);
35
36         final ArrayList<AssemblerOperation> assembler = new ArrayList<>();
37
38         assembler.add(new TextOperation());
39         assembler.add(new P2AlignOperation());
40
41         for (Graph graph : Program.getGraphs()) {
42             BackEdges.enable(graph);
43             graph.walk(new InsertBlockAfterConditionVisitor());
44             BackEdges.disable(graph);
45             assembler.add(new FunctionSpecificationOperation(graph.getEntity().getLdName()));
46         }
47
48         for (Graph graph : Program.getGraphs()) {
49             if (debugRegisterAllocation)
50                 System.out.println(graph.getEntity().getLdName());
51
52             BlockNodesCollectingVisitor collectorVisitor = new BlockNodesCollectingVisitor();
53             graph.walkTopological(collectorVisitor);
54
55

```