class Parent {
    static int field1 = 10;
    static int field2;
    int field3 = 7;
    int field4;

    static void test1() {
    }
    static void test2() {
    }
    void test3() {
    }
    void test4() {
    }
}

class Child extends Parent {
    static int field2 = 1;
    int field4;
    int field5 = 999999;
    static void test2() {
    }
    void test4() {
    }
    void test5() {
    }
    static void main(String[] args) {
    }
}

Now we are ready to invoke main(...)!
call R7[24]

Which invokes the method at 0x720 (the stub!), passing in a hidden parameter "Child.main(String[])V"
1) invokes the JIT compiler that uses the parameter to find Child.main bytecode and compile it
2) Stores the compiled native code for main in memory
3) Updates main's entry in the statics table with this address - so if main was called a second time, it jump to this address instead of the stub address (which here is 0x720)