默安科技
企业信赖的安全伙伴

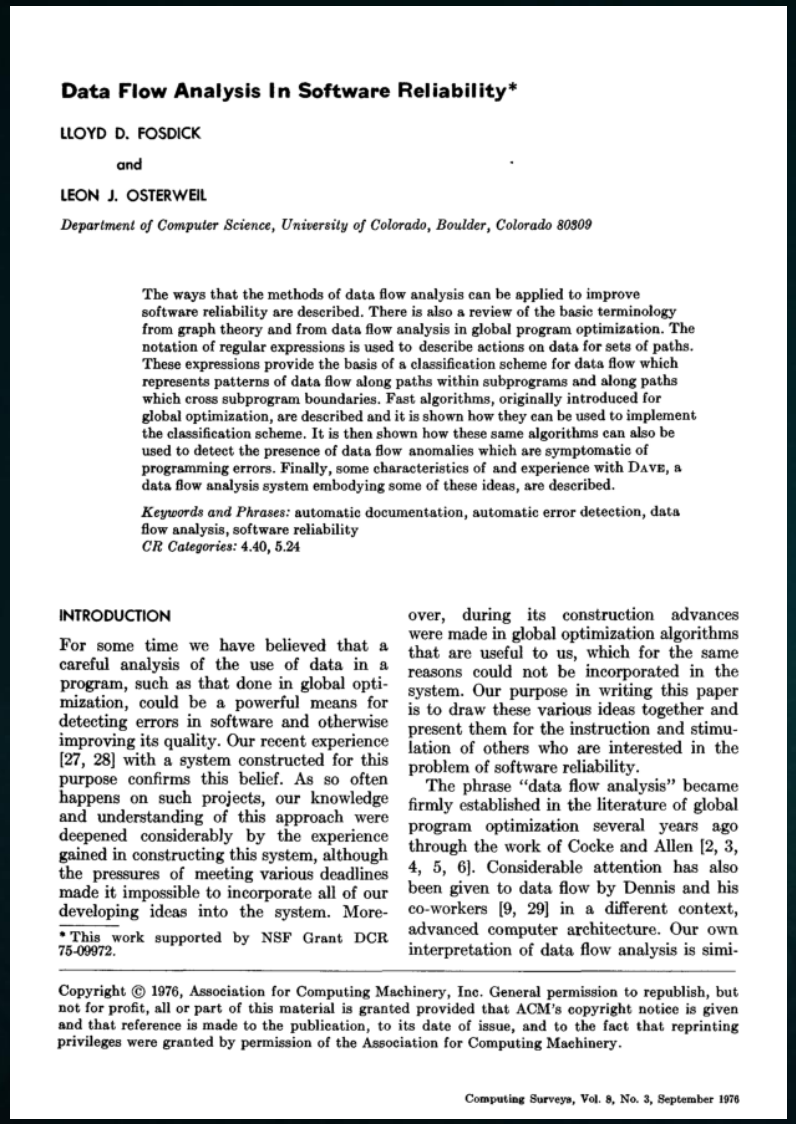
基于虚拟机技术的静态代码审计系统内幕揭秘

默安科技 前沿研究部 郑斯碟

目录

Catalog

默安科技
企业信赖的安全伙伴



Data Flow Analysis In Software Reliability*

LLOYD D. FOSDICK

and

LEON J. OSTERWEIL

Department of Computer Science, University of Colorado, Boulder, Colorado 80309

The ways that the methods of data flow analysis can be applied to improve software reliability are described. There is also a review of the basic terminology from graph theory and from data flow analysis in global program optimization. The notation of regular expressions is used to describe actions on data for sets of paths. These expressions provide the basis of a classification scheme for data flow which represents patterns of data flow along paths within subprograms and along paths which cross subprogram boundaries. Fast algorithms, originally introduced for global optimization, are described and it is shown how they can be used to implement the classification scheme. It is then shown how these same algorithms can also be used to detect the presence of data flow anomalies which are symptomatic of programming errors. Finally, some characteristics of and experience with DAVE, a data flow analysis system embodying some of these ideas, are described.

Keywords and Phrases: automatic documentation, automatic error detection, data flow analysis, software reliability
CR Categories: 4.40, 5.24

INTRODUCTION

For some time we have believed that a careful analysis of the use of data in a program, such as that done in global optimization, could be a powerful means for detecting errors in software and otherwise improving its quality. Our recent experience [27, 28] with a system constructed for this purpose confirms this belief. As so often happens on such projects, our knowledge and understanding of this approach were deepened considerably by the experience gained in constructing this system, although the pressures of meeting various deadlines made it impossible to incorporate all of our developing ideas into the system. Moreover, during its construction advances were made in global optimization algorithms that are useful to us, which for the same reasons could not be incorporated in the system. Our purpose in writing this paper is to draw these various ideas together and present them for the instruction and stimulation of others who are interested in the problem of software reliability.

The phrase "data flow analysis" became firmly established in the literature of global program optimization several years ago through the work of Cocke and Allen [2, 3, 4, 5, 6]. Considerable attention has also been given to data flow by Dennis and his co-workers [9, 29] in a different context, advanced computer architecture. Our own interpretation of data flow analysis is simi-

* This work supported by NSF Grant DCR 75-09972.

Soot

白盒审计漫谈

FortifySCA

Checkmarx

CodeSecure

Data Flow Analysis in Software Reliability

ACM Computing Surveys

Lloyd D. Fosdick & Leon J. Osterweil

数据流分析

状态机系统

边界检测

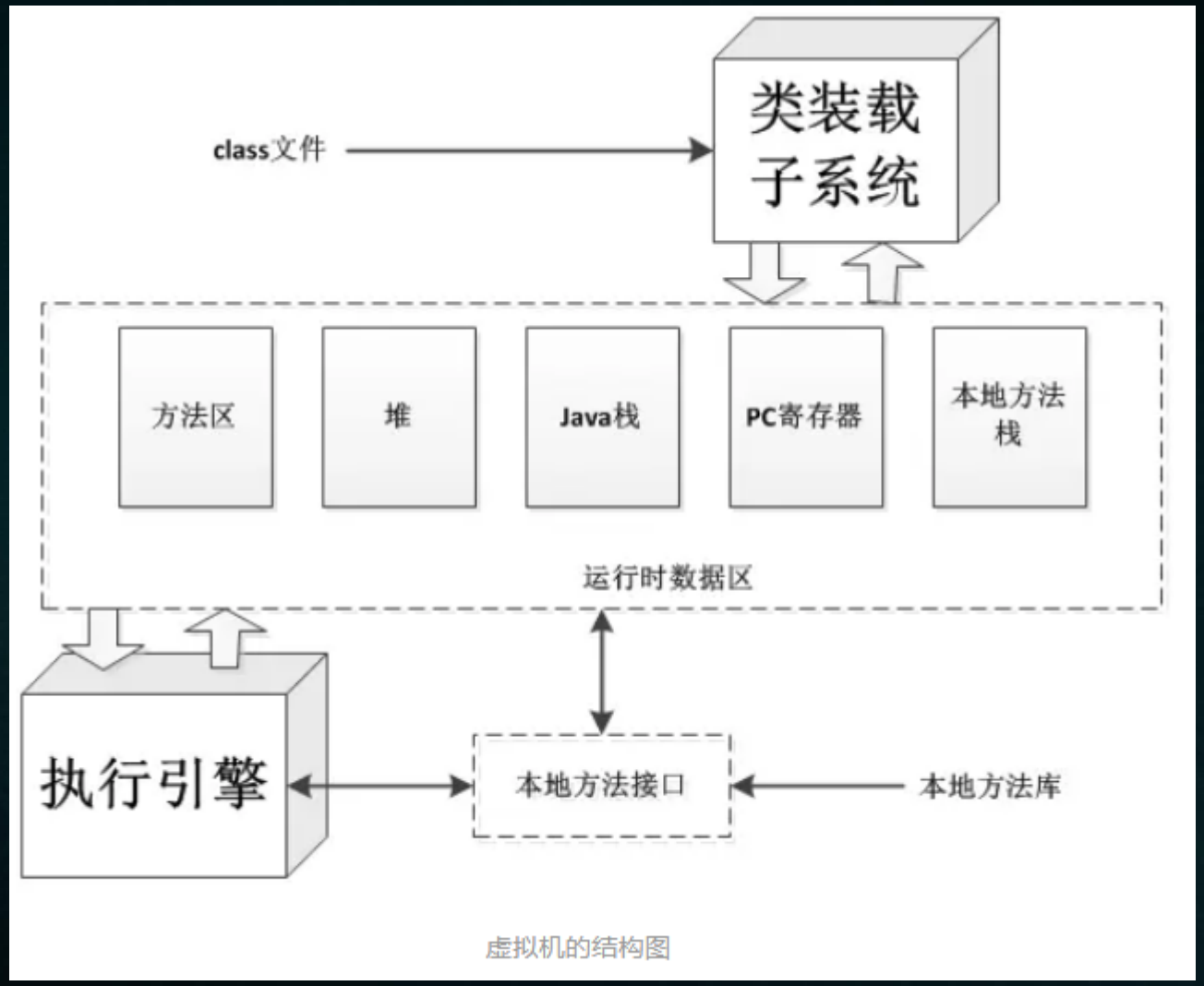
数据类型验证

控制流分析

基于有限状态自动机的静态代码污点追踪的实现

JVM结构图

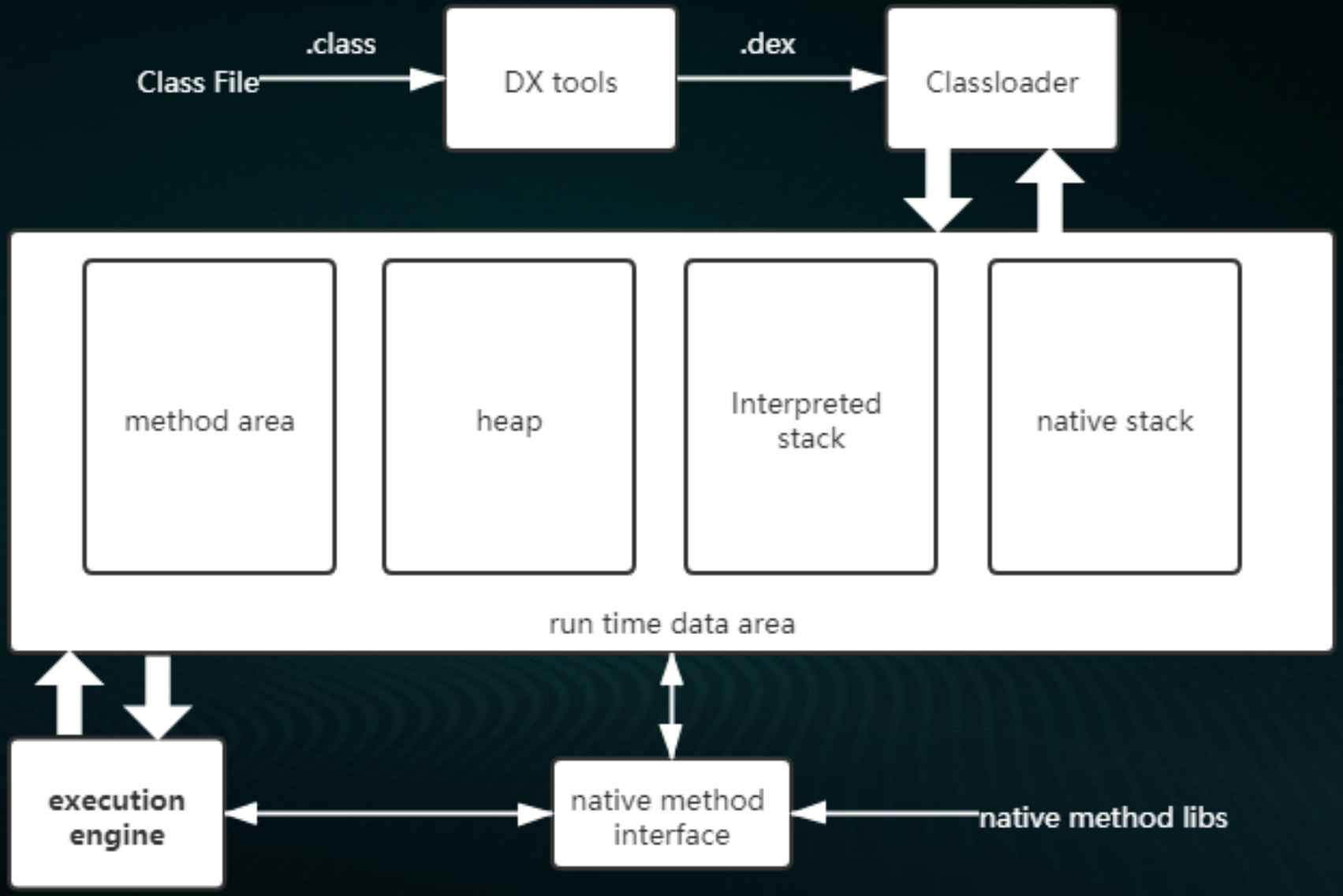
Add title here



虚拟机的结构图

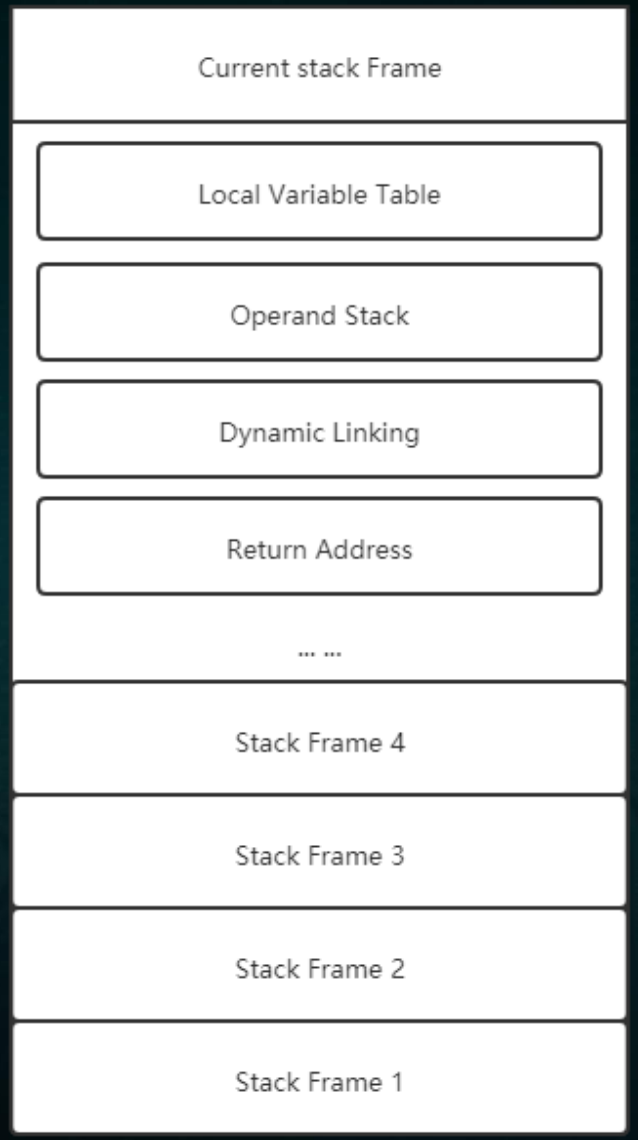
DVM结构图

Add title here

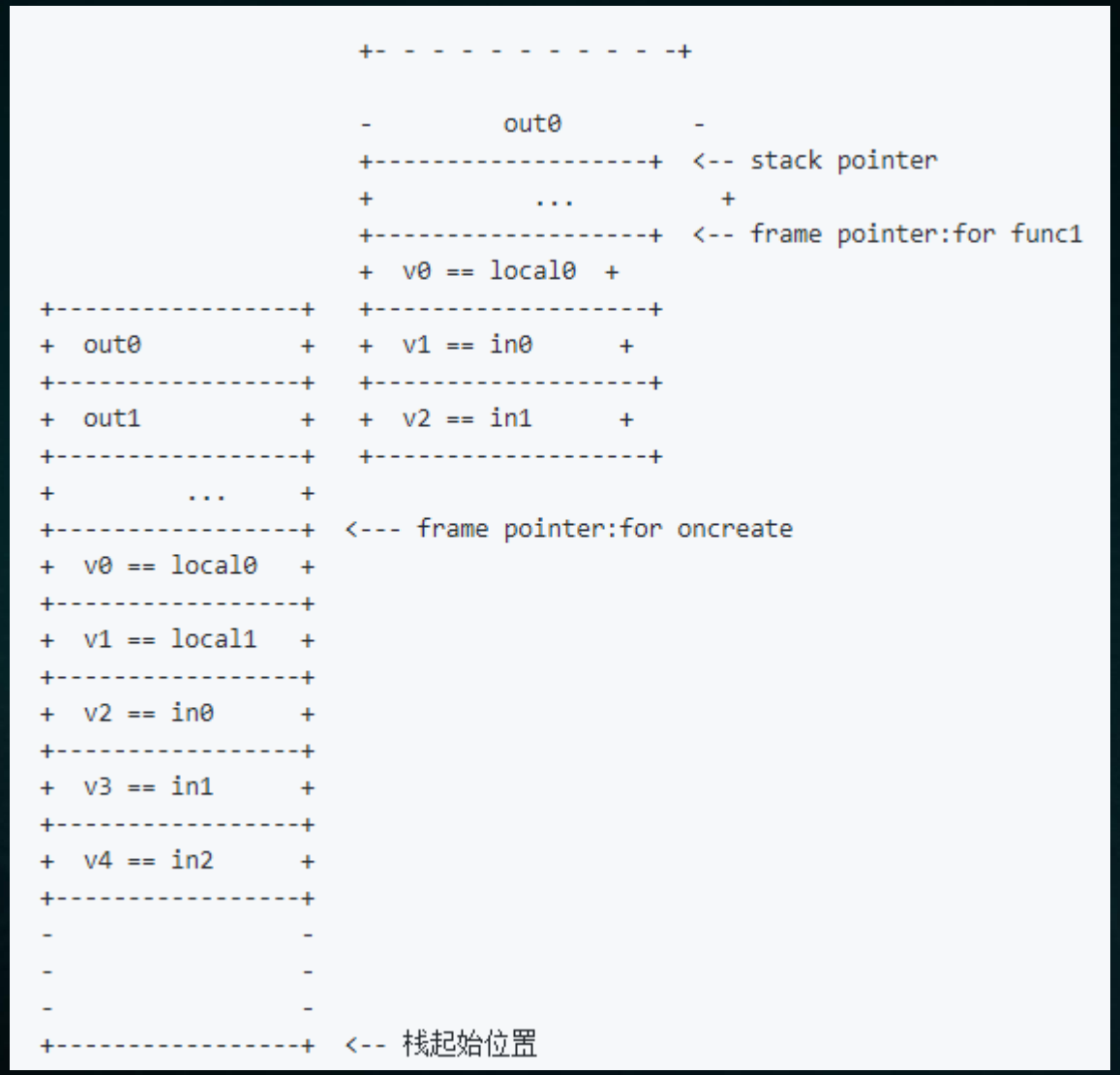


JVM栈结构说明

默安科技
企业信赖的安全伙伴



DVM栈结构说明


out0
<-- stack pointer
...
<-- frame pointer:for func1
v0 == local0
v1 == in0
v2 == in1
out0
out1
...
v0 == local0
v1 == local1
v2 == in0
v3 == in1
v4 == in2
<--- frame pointer:for oncreate
<-- 栈起始位置

DVM&JVM字节码的对比讲解

默安科技
企业信赖的安全伙伴

smali 字节码

```
0: const-string v0, ""
1: const-string v0, "nicejob"
2: invoke-virtual {p1, v0}, Ljava/lang/String;-
>equals(Ljava/lang/Object;)Z
3: move-result v0
4: if-eqz v0, :cond_11
5: sget-object v0, Ljava/lang/System;-
>out:Ljava/io/PrintStream;
6: const-string v1, "Hello Hades"
7: invoke-virtual {v0, v1},
Ljava/io/PrintStream;-
>println(Ljava/lang/String;)V
8: cond_11
9: return-object p1
```

JVM字节码

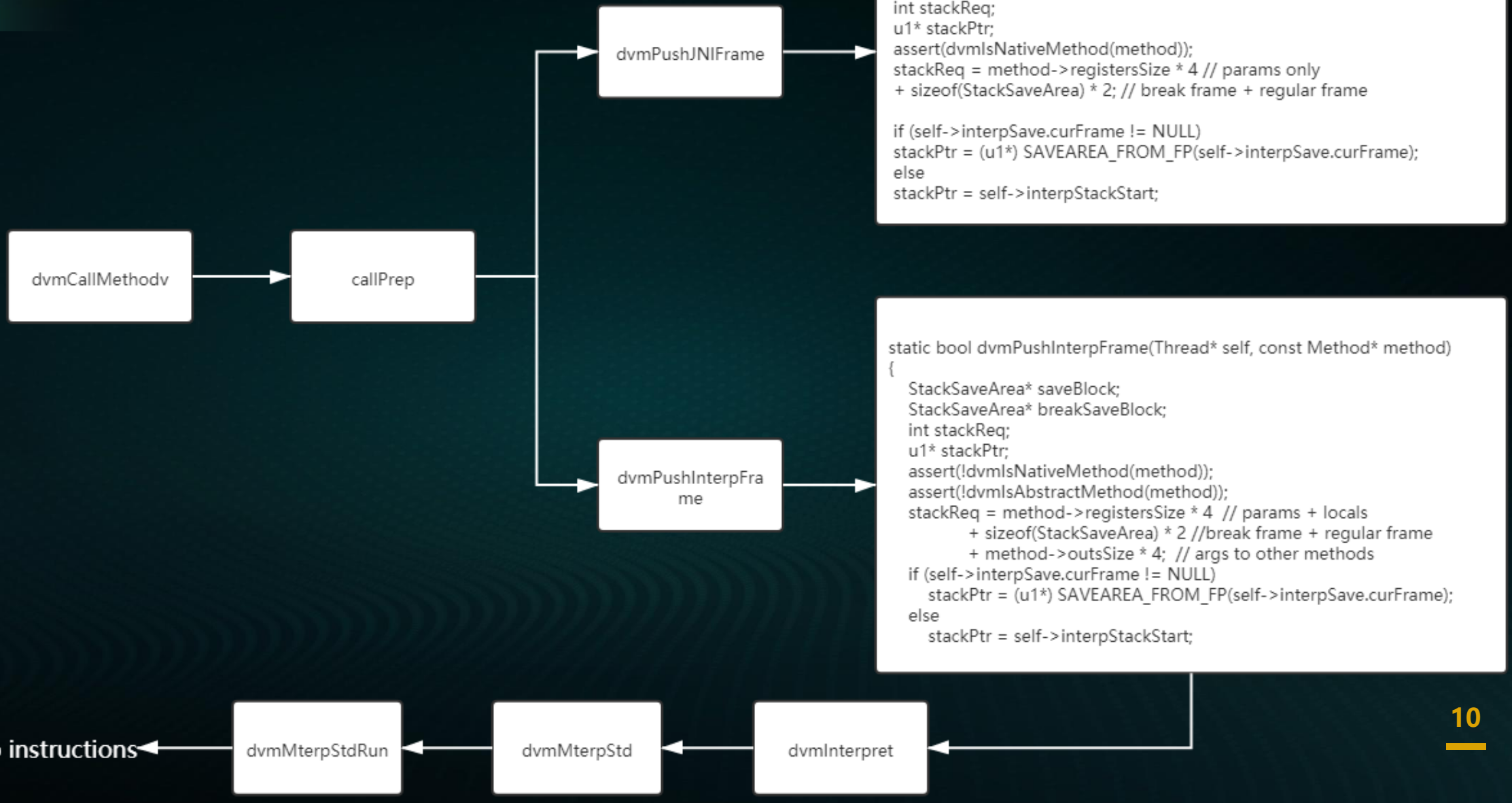
```
0: ldc           #2 // String ""
2: astore_2
3: aload_1
4: ldc           #3 // String nicejob
6: invokevirtual #4 // method def 4
9: ifeq          22
12: aload_1
13: astore_2
14: getstatic     #5// method def 5
17: ldc           #6// String Hello Hades
19: invokevirtual #7// method def 7
22: aload_1
23: areturn
```

Java 源码

```
0: String Msg="";
1: if (p1.equals("nicejob")) {
2:       Msg = p1;
3:       System.out.println("Hello Hades");}
4: return p1;
```

DVM bytecode&JVM bytecode&sourcecode

DVM栈帧分配及指令解释执行




```
bool dvmPushJNIFrame(Thread* self, const Method* method)
{
 StackSaveArea* saveBlock;
 StackSaveArea* breakSaveBlock;
 int stackReq;
 u1* stackPtr;
 assert(dvmIsNativeMethod(method));
 stackReq = method->registersSize * 4 // params only
 + sizeof(StackSaveArea) * 2; // break frame + regular frame

 if (self->interpSave.curFrame != NULL)
 stackPtr = (u1*) SAVEAREA_FROM_FP(self->interpSave.curFrame);
 else
 stackPtr = self->interpStackStart;
```

```
static bool dvmPushInterpFrame(Thread* self, const Method* method)
{
    StackSaveArea* saveBlock;
    StackSaveArea* breakSaveBlock;
    int stackReq;
    u1* stackPtr;
    assert(!dvmIsNativeMethod(method));
    assert(!dvmIsAbstractMethod(method));
    stackReq = method->registersSize * 4  // params + locals
             + sizeof(StackSaveArea) * 2 //break frame + regular frame
             + method->outsSize * 4;  // args to other methods
    if (self->interpSave.curFrame != NULL)
        stackPtr = (u1*) SAVEAREA_FROM_FP(self->interpSave.curFrame);
    else
        stackPtr = self->interpStackStart;
```

DVM opc

根目录: / dalvik / vm / mterp / armv5te

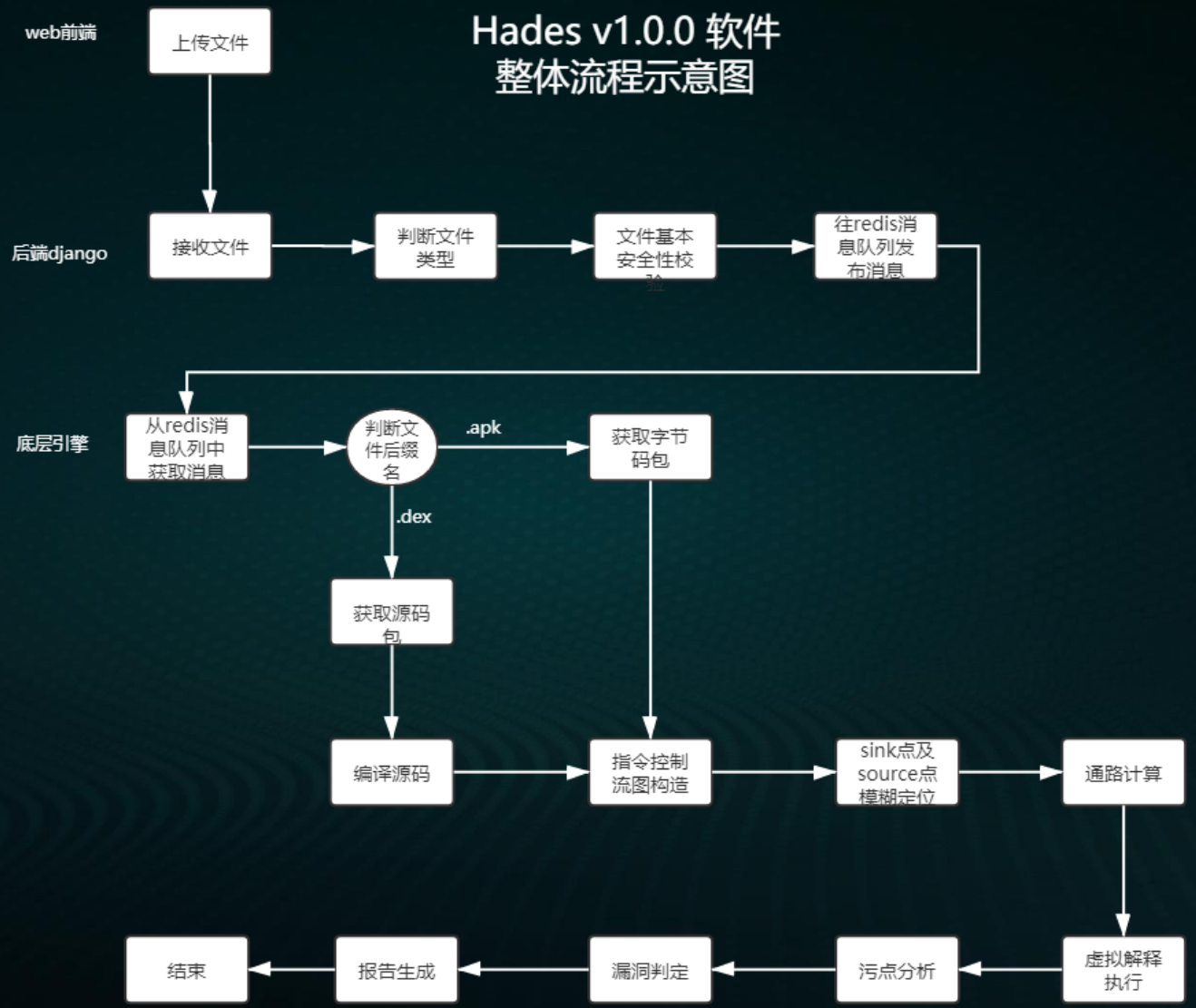
0个文件夹，276个文件　　♡收藏此目录

..

OP_IGET_SHORT.S	181 B	下载 阅读
OP_IPUT_WIDE.S	2.05 KB	下载 阅读
OP_SHL_LONG_2ADDR.S	1.18 KB	下载 阅读
OP_APUT_BYTE.S	80 B	下载 阅读
OP_INVOKE_VIRTUAL_RANGE.S	95 B	下载 阅读
binop.S	1.66 KB	下载 阅读
OP_MONITOR_ENTER.S	743 B	下载 阅读
OP_IPUT_QUICK.S	813 B	下载 阅读
OP_SGET_OBJECT.S	48 B	下载 阅读
OP_XOR_INT_2ADDR.S	82 B	下载 阅读
OP_DIV_FLOAT_2ADDR.S	84 B	下载 阅读
OP_MOVE_FROM16.S	503 B	下载 阅读

Hades 开源项目整体流程示意图

默安科技
企业信赖的安全伙伴



What is CFG

A control flow graph is a representation of a program that makes certain analyses (including dataflow analyses) easier

What information can we get from the CFG?

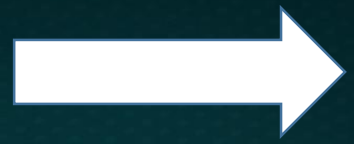
What can we do with CFG

How to build a CFG

```
public class Main {
    public static void main(String[] args) {
        for (int j = 0; j < 5; j++) {
            if(j==1){
                System.out.println("Hello Hades!");
            }
        }
    }
}
```

```
.line 5
const/4 v0, 0x0
:goto_1
const/4 v1, 0x5
if-ge v0, v1, :cond_11
.line 6
const/4 v1, 0x1
if-ne v0, v1, :cond_e
.line 7
sget-object v1, Ljava/lang/System;->out:Ljava/io/PrintStream;
const-string v2, "Hello Hades!"
invoke-virtual {v1, v2}, Ljava/io/PrintStream;->println(Ljava/lang/String;)V
.line 5
:cond_e
add-int/lit8 v0, v0, 0x1
goto :goto_1
.line 10
:cond_11
return-void
```

```
.line 5
const/4 v0, 0x0
:goto_1
const/4 v1, 0x5
if-ge v0, v1, :cond_11
.line 6
const/4 v1, 0x1
if-ne v0, v1, :cond_e
.line 7
sget-object v1, Ljava/lang/System;->out:Ljava/io/PrintStream;
const-string v2, "Hello Hades!"
invoke-virtual {v1, v2}, Ljava/io/PrintStream;->println(Ljava/lang/String;)V
.line 5
:cond_e
add-int/lit8 v0, v0, 0x1
goto :goto_1
.line 10
:cond_11
return-void
```



(b1, b2)
(b2, b6)
(b2, b3)
(b3, b4)
(b3, b5)
(b4, b5)
(b5, b2)

b4

```
.line 7
sget-object v1, Ljava/lang/System;->out:Ljava/io/PrintStream;
const-string v2, "Hello Hades!"
invoke-virtual {v1, v2}, Ljava/io/PrintStream;->println(Ljava/lang/String;)V
```

b2

```
:goto_1
const/4 v1, 0x5
if-ge v0, v1, :cond_11
```

b6

```
.line 10
:cond_11
return-void
```

b3

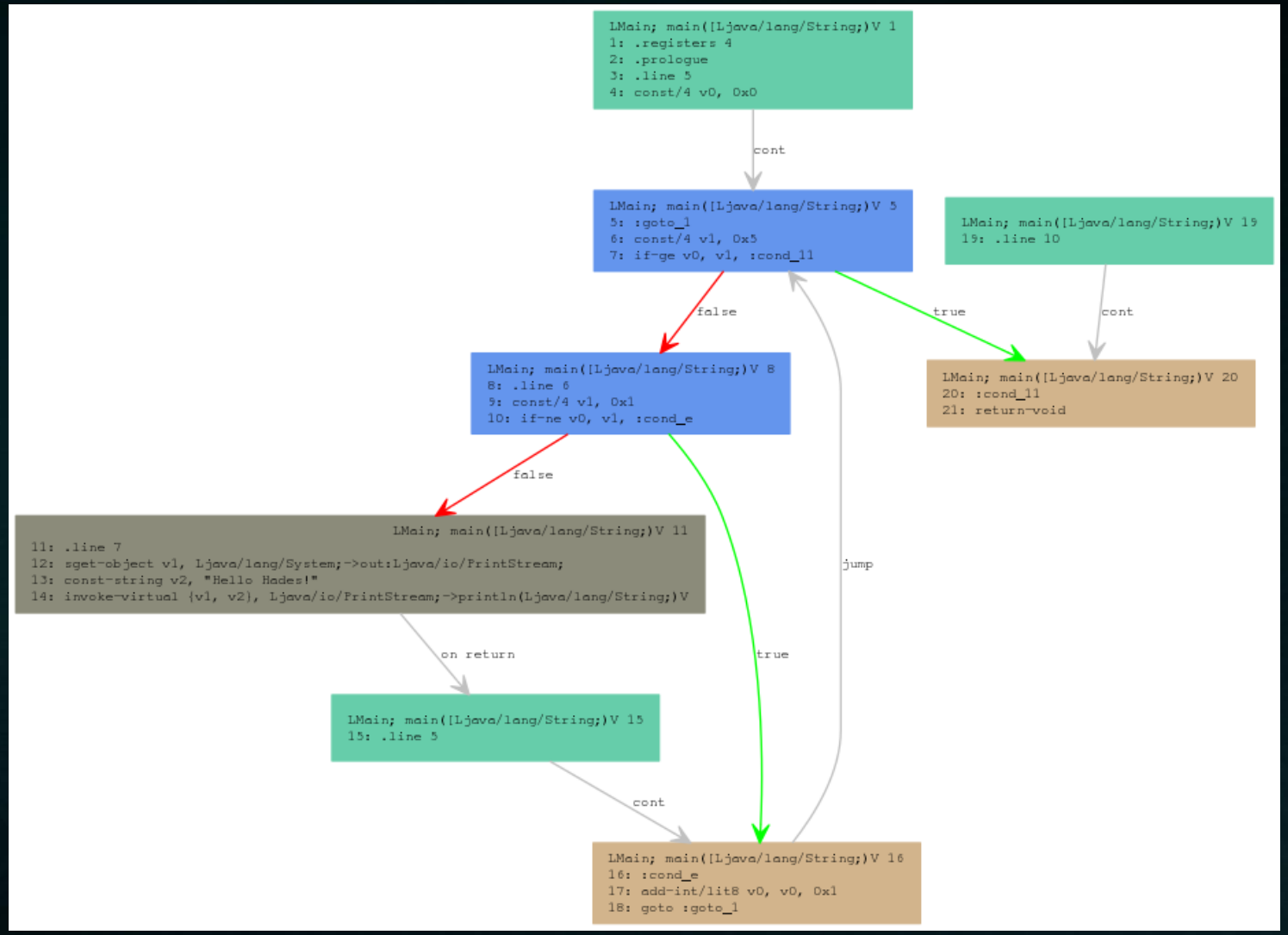
```
.line 6
const/4 v1, 0x1
if-ne v0, v1, :cond_e
```

b1

```
.line 5
const/4 v0, 0x0
```

b5

```
.line 5
:cond_e
add-int/lit8 v0, v0, 0x1
goto :goto_1
```


LMain; main([Ljava/lang/String;)V 1
1: .registers 4
2: .prologue
3: .line 5
4: const/4 v0, 0x0
cont
LMain; main([Ljava/lang/String;)V 5
5: :goto_1
6: const/4 v1, 0x5
7: if-ge v0, v1, :cond_11
false
true
LMain; main([Ljava/lang/String;)V 19
19: .line 10
cont
LMain; main([Ljava/lang/String;)V 8
8: .line 6
9: const/4 v1, 0x1
10: if-ne v0, v1, :cond_e
false
LMain; main([Ljava/lang/String;)V 20
20: :cond_11
21: return-void
LMain; main([Ljava/lang/String;)V 11
11: .line 7
12: sget-object v1, Ljava/lang/System;->out:Ljava/io/PrintStream;
13: const-string v2, "Hello Hades!"
14: invoke-virtual {v1, v2}, Ljava/io/PrintStream;->println(Ljava/lang/String;)V
jump
true
on return
LMain; main([Ljava/lang/String;)V 15
15: .line 5
cont
LMain; main([Ljava/lang/String;)V 16
16: :cond_e
17: add-int/lit8 v0, v0, 0x1
18: goto :goto_1

Path Calculation

sink and source

Memory simulation

Interpreting

Tainted track

过程内信息流跟踪&污点分析

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args|False	
v0	None|False	
v1	None|False	
v2	None|False	
v3	None|False	
v4	None|False	
v5	None|False	

heap

value
tainFlag

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	None	False
v1	None	False
v2	None	False
v3	None	False
v4	None	False
v5	0x2	False

value
tainFlag

heap

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

Find source

register List

name	value	taintFlag
p0	args	False
v0	None	False
v1	None	False
v2	None	False
v3	None	False
v4	None	False
v5	0x2	False

heap

value
tainFlag

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	None	False
v1	None	True
v2	None	False
v3	None	False
v4	None	False
v5	0x2	False

heap

value
tainFlag

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	None	False
v1	None	True
v2	None	False
v3	None	False
v4	0xa	False
v5	0x2	False

heap

value
tainFlag

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	False
v3	None	False
v4	0xa	False
v5	None	False

heap

value
tainFlag

None	None	None	None	None	None	None	None	None	None
False	False	False	False	False	False	False	False	False	False

arr0

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	True
v3	None	False
v4	0xa	False
v5	0x2	False

heap

value	None	None	None	None	None	None	None	None	None	None	arr0
tainFlag	False	False	False	False	False	False	False	False	False	False	

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	True
v3	None	True
v4	0xa	False
v5	0x2	False

heap

value	None	None	None	None	None	None	None	None	None	None	arr0
tainFlag	False	False	False	False	False	False	False	False	False	False	

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	True
v3	None	True
v4	0xa	False
v5	0x2	False

heap

value
tainFlag

None False	None False	None False	None False	None False	None False	None False	None True	None False	None False

arr0

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	True
v3	None	True
v4	None	True
v5	0x2	False

heap

value	None	None	None	None	None	None	None	None	None	None	arr0
tainFlag	False	False	False	False	False	False	False	True	False	False	

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->responseToHTML(Ljava/lang/String;)V
    return-void
.end method
```

Vulnerability

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	True
v3	None	True
v4	None	True
v5	0x2	False

heap

value	None	None	None	None	None	None	None	None	None	None	arr0
tainFlag	False	False	False	False	False	False	False	True	False	False	

过程间信息流跟踪&污点分析

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args|False	
v0	None|False	
v1	None|False	
v2	None|False	
v3	None|False	
v4	None|False	
v5	None|False	

heap

value
tainFlag

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args|False	
v0	None|False	
v1	None|False	
v2	None|False	
v3	None|False	
v4	None|False	
v5	0x2|False	

heap

value
tainFlag

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

Find source

register List

name	value	taintFlag
p0	args	False
v0	None	False
v1	None	False
v2	None	False
v3	None	False
v4	None	False
v5	0x2	False

heap

value
tainFlag

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	None	False
v1	None	True
v2	None	False
v3	None	False
v4	None	False
v5	0x2	False

heap

value
tainFlag

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args|False	
v0	None|False	
v1	None|True	
v2	None|False	
v3	None|False	
v4	0xa|False	
v5	0x2|False	

heap

value
tainFlag

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	False
v3	None	False
v4	0xa	False
v5	0x2	False

heap

value	None	None	None	None	None	None	None	None	None	None	arr0
tainFlag	False	False	False	False	False	False	False	False	False	False	

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	True
v3	None	False
v4	0xa	False
v5	0x2	False

heap

value	None	None	None	None	None	None	None	None	None	None	arr0
tainFlag	False	False	False	False	False	False	False	False	False	False	

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args|False	
v0	arr0Ref|False	
v1	None|True	
v2	None|True	
v3	None|True	
v4	0xa|False	
v5	0x2|False	

heap

value tainFlag	None False	None False	None False	None False	None False	None False	None False	None False	None False	None False	arr0

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	True
v3	None	True
v4	0xa	False
v5	0x2	False

heap

value
tainFlag

None False	None False	None False	None False	None False	None False	None False	None True	None False	None False

arr0

```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	True
v3	None	True
v4	None	True
v5	0x2	False

heap

value	None	None	None	None	None	None	None	None	None	None	arr0
tainFlag	False	False	False	False	False	False	False	True	False	False	

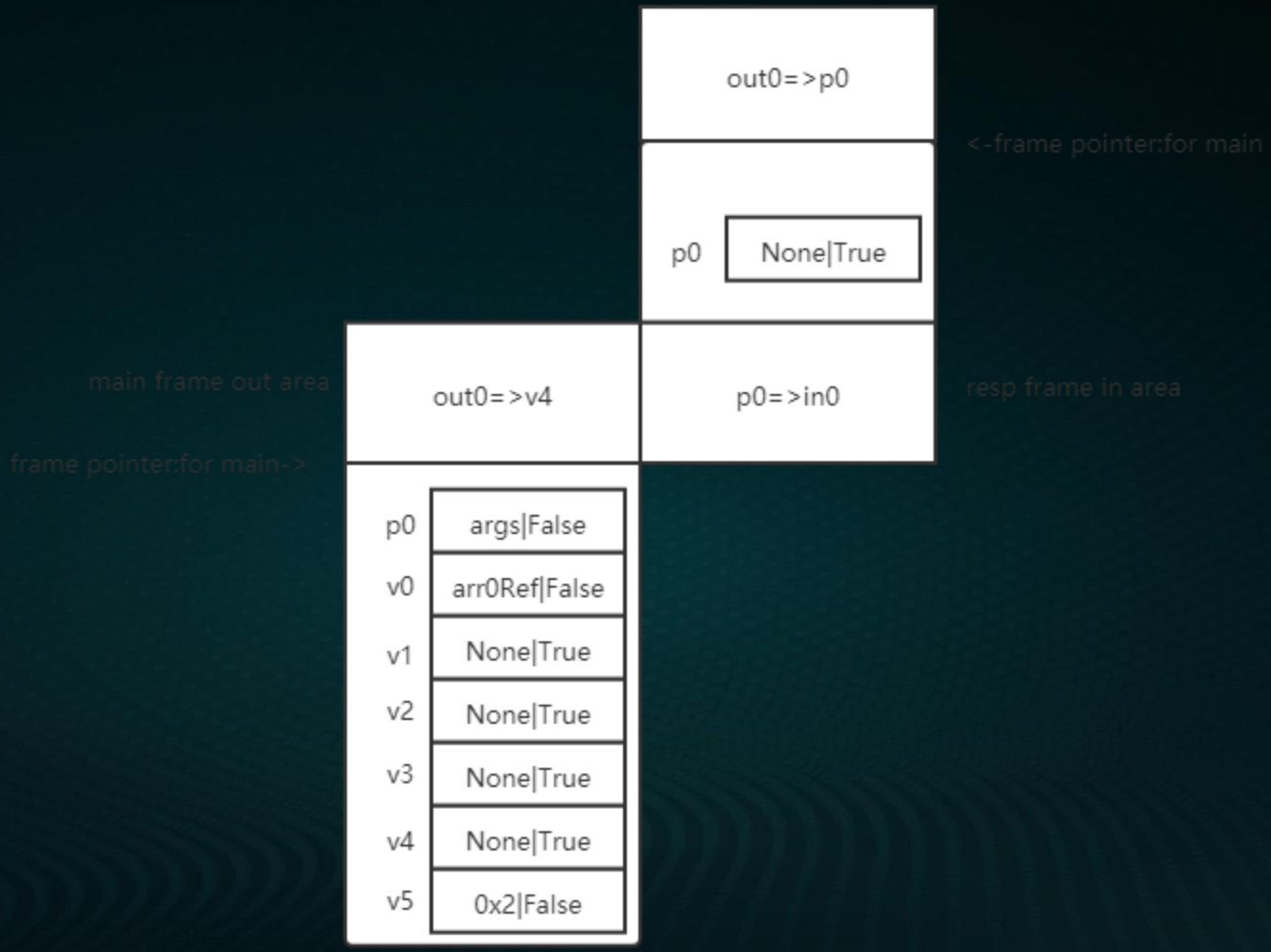
```
.method public static main([Ljava/lang/String;)V
    .param p0, "args"    # [Ljava/lang/String;
    const/4 v5, 0x2
    invoke-static {}, LMain;->getParameters()Ljava/lang/String;
    move-result-object v1
    .local v1, "source":Ljava/lang/String;
    const/16 v4, 0xa
    new-array v0, v4, [Ljava/lang/String;
    .local v0, "arr":[Ljava/lang/String;
    move-object v2, v1
    .local v2, "t1":Ljava/lang/String;
    move-object v3, v2
    .local v3, "t2":Ljava/lang/String;
    aput-object v3, v0, v5                    method invoke
    aget-object v4, v0, v5
    invoke-static {v4}, LMain;->resp(Ljava/lang/String;)V
    return-void
.end method
```

register List

name	value	taintFlag
p0	args	False
v0	arr0Ref	False
v1	None	True
v2	None	True
v3	None	True
v4	None	True
v5	0x2	False

heap

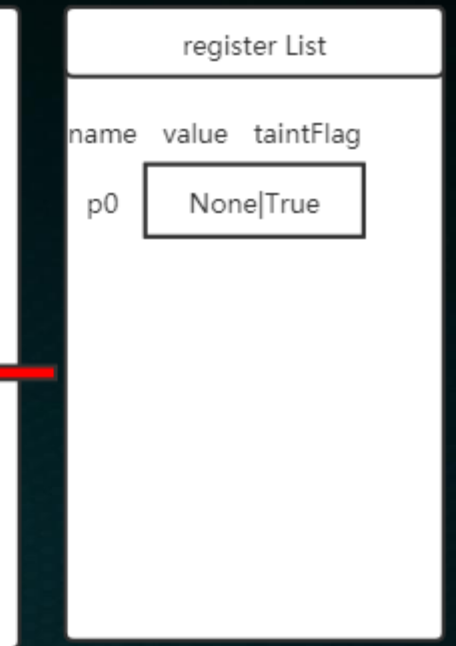
value	None	None	None	None	None	None	None	None	None	None	arr0
tainFlag	False	False	False	False	False	False	False	True	False	False	

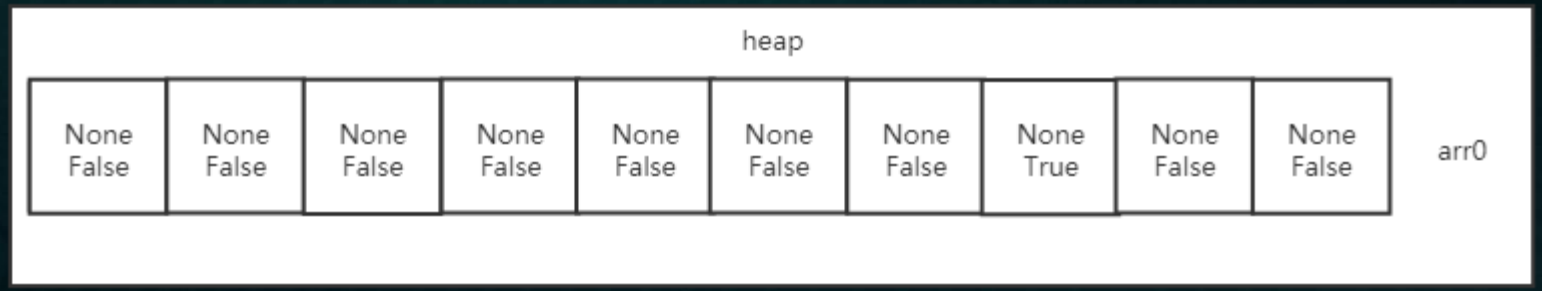

out0=>p0
<-frame pointer:for main
p0
None|True
main frame out area
out0=>v4
p0=>in0
resp frame in area
frame pointer:for main->
p0
args|False
v0
arr0Ref|False
v1
None|True
v2
None|True
v3
None|True
v4
None|True
v5
0x2|False

```
.method public static resp(Ljava/lang/String;)V
    .registers 1
    .param p0, "resp"    # Ljava/lang/String;

    .prologue                                          Vulnerability
    .line 12
    invoke-static {p0}, LMain;->responseToHTML(Ljava/lang/String;)V

    .line 13
    return-void
.end method
```





问题一:说出两种图的遍历算法

问题二:Hades的中间语言是什么?

问题三:Smali字节码和JVM字节码的区别是?

默安科技
企业信赖的安全伙伴

THANK YOU