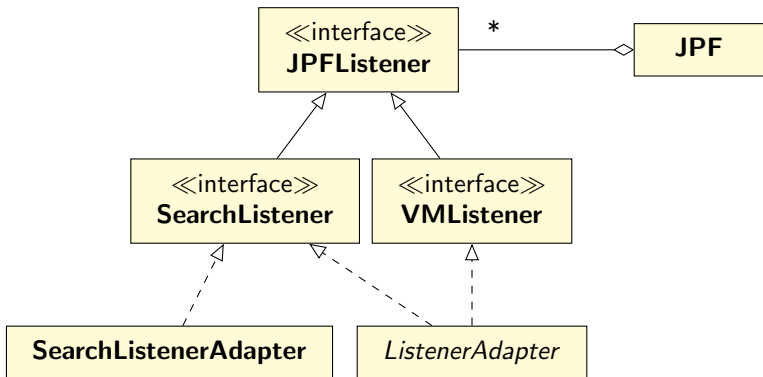


Listen
EECS 4315

www.eecs.yorku.ca/course/4315/





```
public interface VMListener extends JPFListener {
    // VM has been initialized and, hence, classes
    // have been loaded
    void vmInitialized(VM vm);

    // A number of methods related to the execution
    // of instructions
    void executeInstruction(VM vm,
        ThreadInfo currentThread,
        Instruction instructionToExecute);

    void instructionExecuted (VM vm,
        ThreadInfo currentThread,
        Instruction nextInstruction,
        Instruction executedInstruction);
    ...
}
```

```
// A number of methods related to threads
void threadStarted(VM vm,
    ThreadInfo startedThread);
...

// Class has been loaded
void loadClass(VM vm,
    ClassFile classFile);

// A number of methods related to objects
void objectCreated(VM vm,
    ThreadInfo currentThread,
    ElementInfo newObject);
...
```

```
// A number of methods related to garbage
// collection
void gcBegin(VM vm);
...

// A number of methods related to exceptions
void exceptionThrown(VM vm,
    ThreadInfo currentThread,
    ElementInfo thrownException);
...
```

```
// A number of methods related to choice
// generators
void choiceGeneratorRegistered(VM vm,
    ChoiceGenerator<?> nextCG,
    ThreadInfo currentThread,
    Instruction executedInstruction);
...

// A number of methods about methods
void methodEntered(VM vm,
    ThreadInfo currentThread,
    MethodInfo enteredMethod);
...
}
```

Write a listener that prints a * whenever the garbage collector is invoked by JPF.

Write a listener that measures the amount of time (in milliseconds) JPF's garbage collector takes whenever it is invoked by JPF.

Implement a listener which prints the mnemonics of the bytecode instructions that are executed by JPF.

Implement a listener which prints a method that is invoked most often by JPF and the number of times that it is invoked.