Static Analysis and Software Verification
TD1 : Introduction

Nga Nguyen

Question 1 : Try to detect all possible errors in this Java class :

```
import java.io.*;
public class Foo{
   private byte[] b;
   private int length;
   Foo(){
      length = 40;
      b = new byte[length];
   }
   public void bar(){
      int y;
      try {
         FileInputStream x = new FileInputStream("z");
         x.read(b,0,length);
         x.close();
      }
      catch(Exception e){
         System.out.println("Oopsie");
      }
      for(int i = 1; i <= length; i++){
         if (Integer.toString(50) == Byte.toString(b[i]))
         System.out.print(b[i] + " ");
      }
   }
}
```

Question 2 : Here is a C function computing Fibonacci numbers

```
unsigned int fib(unsigned int m) {
   unsigned int f0 = 0, f1 = 1, f2, i;
   if (m <= 1) {
      return m;
   }
   else {
      for (i = 2; i <= m; i++) {
         f2 = f0 + f1;
         f0 = f1;
         f1 = f2;
      }
      return f2;
   }
}
```

(a) build a control-flow graph of this function, instruction by instruction ;

(b) modify the above graph to have basic blocs.

Question 3 : Install and test the following static analyzers for Java (Checkstyle, FindBugs and PMD) :

— Checkstyle : http ://checkstyle.sourceforge.net/

— FindBugs : http ://findbugs.sourceforge.net/

— PMD : http ://pmd.sourceforge.net/