

## Software Testing: Preparatory work for tutorial on group task 2

### Structural Testing

The code below is part of a method in the `ConvexHull` class in the VMAP system. The following is a small fragment of a method in the `ConvexHull` class. For the purposes of this exercise you do not need to know the intended function of the method. The parameter  $p$  is a Vector of Point objects, `p.size()` is the size of the vector  $p$ , `(p.get(i)).x` is the  $x$  component of the  $i^{\text{th}}$  point appearing in  $p$ , similarly for `(p.get(i)).y`. This exercise is concerned with structural testing of code and so the focus is on creating test sets that satisfy some particular coverage criterion.

```
Vector doGraham(Vector p) {
    int i,j,min,M;

    Point t;
    min = 0;

    // search for minimum:
    for(i=1; i < p.size(); ++i) {
        if( ((Point) p.get(i)).y <
            ((Point) p.get(min)).y )
        {
            min = i;
        }
    }

    // continue along the values with same y component
    for(i=0; i < p.size(); ++i) {
        if( ((Point) p.get(i)).y ==
            ((Point) p.get(min)).y ) &&
            (((Point) p.get(i)).x >
             ((Point) p.get(min)).x ))
        {
            min = i;
        }
    }
}
```

- **Prerequisites:** before the tutorial you should review the reading on structural testing. In particular, you should read Pezzè and Young's chapter 12 which will provide adequate background for this tutorial.
- **Preparation:** Review the code fragment drawn from the `doGraham` method above. If need be, check the documentation on the Vector class and any other Java documentation you might require.

## Activity

Having considered this code fragment you should do the following preparatory activity, writing brief notes as described and submit them to the designated tutor. The tutorial session will review this work and make suggestions on how to tackle the second group task.

1. Convert the Java code comprising the beginning of the `doGraham` method into a flow graph. Perhaps split the group into people who make the flow graph and people who check it is correct. **You can either use a drawing package to draw this or scan a reasonably neatly hand-drawn diagram.**
2. Construct test sets for your flowgraph that are adequate for the following criteria (**you should write your tests as vectors of pairs of numbers (i.e. lists of points), for example `[]` (the empty vector, `[(0,0),(-1,0),(-2,-3)]`, ... together with your expected results):
  - Statement Coverage.
  - Branch Coverage.**
3. **You should send your flowgraph and test sets to the designated tutor for this task.**
4. The first task in the tutorial meeting will be for the two groups to swap test sets and check their adequacy.