CS427 Quiz 5

December 3, 2015

Name: ____________________________
Netid: ____________________________

This is a closed book, closed notes quiz. You may not use calculators or any other electronic devices. Any sort of cheating on the quiz will result in a zero grade.

This quiz has two pages.

We cannot give any clarifications about the quiz questions. If you are unsure of the meaning of a specific question, write down your assumptions and proceed to answer the question on that basis.

1. Describe two reasons why XP advocates having a *spike solution* before diving into a project.

Solution:
We gave credit to answers related to better estimating the user stories’ work, getting familiar with the technology, doing risk management (seeing if the work is feasible or not, not throwing everyone in on an idea before it’s settled). We did not give points to improving testing, learning what the project should be like (that kind of planning does not need spike solution), communication between people, having some backup artifact (it’s expected to be thrown away anyway).

2. Write a postcondition for the method below. Fill in the blank with a boolean expression.

    // returns the absolute value of a number.
    public static int abs(int number)
    {
        if(number < 0)
```c
{
    number = number * -1;
}
assert ____________________;
return number;
}
```

**Solution:**
The common answer was “assert number>=0”, and we gave full credit for that.

3. The testing catalog from the required reading has some suggestions for testing input in various situations. Say you want to test some linked structures (trees, queues, graphs, etc). List two types of structures from the reading that a developer should test.

**Solution:**
empty structure, minimal non-empty structure (singleton, one element), circular structure, depth greater than one (did not accept really big ones because reason given was for performance); these are listed in the short version of the Testing Catalog

4. The slides on testing (8u.pptx) described condition coverage, branch coverage, and coverage subsumption. Does condition coverage subsume branch coverage? Why or why not?

**Solution:**
Yes condition coverage subsumes branch coverage (i.e., if a test suite achieves 100% condition coverage then it always achieves 100% branch coverage as well).

---

Quiz № 5