Program Regrade Request
Attach to front of graded hardcopy
Be complete but concise

Name: 

Date: 

Assignment: 

Class: 

1. (Yes/No) I understand that I should only request regrading when I have a credible justification. This is not an opportunity to request a regrade of an assignment just to see if it might help my grade. I understand that abuse of the regrade request may result in a lower grade.

2. (Fill in the blank e.g., love, despise) I ____________________ fluffy bunnies and myself as a person.

3. (Yes/No) I have carefully read the test document and executed the full test on my original program submission. I conducted my tests on one of the department Linux boxes (unless otherwise specified) using all compile/JVM options given in the test document. Below are all additional special options, etc. that I used to execute the tests.

4. Clearly list the specific (sub)tests to be regraded. Failure to list the individual (sub)tests will result in rejection. Only request regrading where it can be well justified. Carefully read the grounds for regrading below. For each requested (sub)test, make sure you carefully provide the grounds and supporting evidence:
There are two grounds for regrading a (sub)test:

**Correct Solution**
You believe your submission to be correct. You must include a justification(s) for this claim. Justifications include 1) the grading is simply incorrect according to the test, 2) the specification is flawed (e.g., ambiguous), and/or 3) the test is flawed. Be sure to include your supporting evidence for each claim.

**Reconsideration**
At the time of submission, you believed your solution to be correct and well tested; however, the test document focused on some particular element that you missed (e.g., your program works everyday but Wednesday). The burden is on you to demonstrate all of the following:

1. You heavily tested your code and
2. Some part of the test (which you missed) *disproportionately* impacted your grade.

A disproportionate impact is one where the flaw causes test failures unrelated to the actual flaw. For example, let’s say your constructor disallows negative numbers even though you now realize such numbers are allowed by the specification. Tests focused on your constructor should fail. Such tests are not a disproportionate impact so you should not request regrading for such tests. However, what if there are tests focused on something other than the constructor and the tests happen to use negative numbers? Here you are failing tests that are not directly related to the flaw. This may create a disproportionate impact on your grade. To be clear, do not submit reconsideration requests for tests that successfully find the very flaw they are testing. It doesn’t matter that the changes to fix the problem are small.

For reconsideration, you must include all changes to your code required to fix the problem. These changes must be 1) specific and 2) simple. Specific changes give clear and concise directions on what to change and where. Specific changes give clear instructions, not long paragraphs. Simple means the changes can be described in just a few lines. For example,

```
Change the following line
if (number <= 0)
to
if (number < 0)
```

Put yourself in the place of the TA. Could you quickly make and evaluate the changes give your request if you were the TA?

Do not submit your revised code. The TA will only work from what you submitted. If what you submitted isn’t easy to correct, then the change is not simple and, therefore, reconsideration should not be requested.