

Project 1: Adding Source Code and Grading

CSE-150-01

You are encouraged to add new classes to your solution as you see fit; the code we provide you is not a complete skeleton for the project. Your project code will be automatically graded. There are two reasons for this:

1. An autograder can test your code a lot more thoroughly than a TA can, yielding more fair results.
2. An autograder can test your code a lot faster than a TA can.

Of course, there is a tradeoff. Everything that will be tested needs to have a standard interface that the autograder can use, leaving slightly less room for you to be creative. Your code must strictly follow these interfaces (the documented ***Interface** classes).

Since your submissions will be processed by a program, there are some very important things you must do, as well as things you must not do. For **all** the projects in this course:

1. *Do not* modify **Makefile**, **except** in the **test** directory to add source files. When grading, we will be using our own **Makefile** (**javac** automatically finds source files to compile, so we don't need you to submit a **Makefile**).
2. Only modify **nachos.conf** according to the project specifications. When grading, we will be using our own **nachos.conf**. Do not rely on any additional keys being in this file.
3. *Do not* modify any classes in the **nachos.machine** package, the **nachos.ag** package, or the **nachos.security** package. Also, do not add any classes to these packages, as they would not be used during grading.
4. *Do not* add any new packages to your project. All the classes you submit must reside in the packages we provide.
5. *Do not* modify the API for methods that the autograder uses. This is enforced every time you run Nachos by **Machine.checkUserClasses()**. If an assertion fails there, you will know you have modified an interface that needs to stay the way it was given to you.
6. *Do not* directly use Java threads (the **java.lang.Thread** class). The Nachos security manager will not permit it. All the threads you use should be managed by TCB objects (see the documentation for **nachos.machine.TCB**).
7. *Do not* use the **synchronized** keyword in any of your code. We will **grep** for it and reject any submission that contains it.
8. *Do not* directly use Java **File** objects (in the **java.io** package). In later projects, when we start dealing with files, you will use a Nachos file system layer.
9. Sprinkled throughout the Nachos classes there are some mandatory calls to the autograder. Leave them as they are.