

```
drv_bubble(3000.0);
```

3. As noted in point 5 of section 1.3.4, due to a known bug in the Tamer, the generated X10 code may contain multiple definitions of the same function. This needs to be manually fixed by deleting one of the duplicate copies of such a method in the generated X10 file, which otherwise would result in an error while compiling with the X10 compiler. Note that the bubble sort example being used in this guide will *not* have any duplicate method definitions.

The following section gives details on how to get the X10 compiler and how to use it to compile and execute an X10 program both with the Java and the C++ backends.

## 2 Artifact evaluation

Once the X10 code is generated by following the steps in the previous section, it is evaluated for performance by compiling and executing it using the X10 compiler. To evaluate the effects on performance of the various features of the MiX10 compiler, appropriate switches are used to generate code with a particular feature turned on/off, as described in the getting started guide in the previous section. Similarly, different switches are used in the X10 compiler to evaluate the generated code more elaborately.

This section covers the following three topics: (1) Instructions on how to obtain and use the X10 compiler; and (2) An overview of the combinations of the command line switches for evaluating the artifact.

### 2.1 Obtaining and using the X10 compiler

The X10 compiler is freely available under the Eclipse public License. It can be obtained in the form of executable binaries, the source code or an Eclipse-like API including the compiler. The X10 compiler provides two backends, a Java backend and a C++ backend, effectively giving two different compilers.

Follow the instructions below to obtain the X10 compiler and compile and run the generated X10 code using it:

1. **Obtaining the X10 compiler:** The easiest way to get an X10 compiler is to download it in the form a development toolkit, called X10DT, which is an eclipse-like IDE for X10 and includes a builtin X10 compiler. The second way (and the way we will give the usage instructions for) is to get the pre-built binary of the X10 compiler. The third way is to download the X10 compiler source code and compile it yourself. We will also give a brief description of this way as we need it to use the column-major indexing for the X10 arrays described in section 4.1.1 of the paper. All the three forms of the X10 compiler can be obtained from the following URL: <http://x10-lang.org/software/download-x10/latest-release.html>. We used X10 version 2.4.0, but the latest version 2.4.3 should also work.
2. **Using the X10 compiler:** We will describe how to use the X10 compiler downloaded as the pre-built binary version. For the sake of simplicity, we will assume that the downloaded