The Eleventh ICFP Programming Contest: Simulator

To help you test your controller, NASA is providing a simulator. This program is available for several different systems, including the LiveCD environment. To run the server, you must supply it with the name of a map file, so NASA is also providing some sample maps for download.

Running the simulator

For example, the command

```
./server -v map1.wrld
```

will run the server on `map1.wrld` with a graphical view of the simulation. When the simulator starts up, it prints the message:

```
waiting for client connection on port n
```

where *n* is the port number used to connect to the server. When running graphical mode, the simulator can be terminated by typing the `q' key. Assuming your program resides is in the \texttt{bin} directory, you can then run it with the command

```
bin/run hostname n
```

where *hostname* is the name of the machine on which the server is running. You can also specify the port using the `-p` option. For example,

```
./server -p 19023 map1.wrld
```

will run the server in non-graphical mode using port 19023.

Map-file format

NASA has supplied some sample maps for download, but you may wish to define your own.

Maps are represented as JSON (JavaScript Object Notation) files. (See `http://www.json.org/` for details on JSON.) The format of a map file is as follows:

```
{
"size" : INT,
"timeLimit" : INT,
"vehicleParams" : PARAMS,
"martianParams" : PARAMS,
"craters" : [ { "x" : FLOAT, "y" : FLOAT, "r" : FLOAT }, ... ],
"boulders" : [ { "x" : FLOAT, "y" : FLOAT, "r" : FLOAT }, ... ],
"runs" : [ RUN, ... ]
}
```

where *PARAMS* is a JSON object with the following format:

```
{
"maxSpeed" : FLOAT,
"accel" : FLOAT,
"brake" : FLOAT,
"turn" : FLOAT,
"hardTurn" : FLOAT,
"rotAccel" : FLOAT,
```

```
"frontView" : FLOAT,
"rearView" : FLOAT
}
```

and a *RUN* is a JSON object with the following format:

```
{
"vehicle" : VEHICLE,
"enemies" : [ ENEMY, ... ]
}
```

A *VEHICLE* object has the form

```
{
"x" : FLOAT,
"y" : FLOAT,
"dir" : FLOAT
}
```

and an *ENEMY* object is a vehicle object extended with the following extra fields:

```
{
... as per VEHICLE ...
"speed" : FLOAT,
"view" : FLOAT
}
```

Downloads

The current version of the sample server is 1.1. It is available various versions.

File	Size (bytes)	md5 hash	Description
sample-maps.tgz	5516	80f7a0549e0231b9e59e1e273ed9e8a8	Sample maps (v1.1)
livecd.tgz	361588	5968d9d0de1e83fb3064774036923ae4	Server for LiveCD environment (v1.2)
no-gui-static.tgz	526142	ea72e4118ec5c6b4f02d1d16a26b42a0	Sample-server that was statatically linked against the LiveCD (v1.2). Does not have GUI support.
linux2.6.24.tgz	362976	75196741cbc0e17d168cd78097e041c3	Sample server for Linux 2.6.24 (v1.2)
macos10.5-intel.tgz	365139	5cb02e0dc5fd67572391b999e2385841	Sample server for Mac OS X 10.5/Intel (v1.2)
macos10.4-intel.tgz	367272	945a7e8c8980fee1b3422d8e92ac0b26	Sample server for Mac OS X 10.4/Intel (v1.2)
macos10.4-ppc.tgz	439324	d8d4c543c31be10c5d0a83e7055039e3	Sample server for Mac OS X 10.4/PowerPC (v1.2)

Change history

Version 1.0

Initial version.

Maps Version 1.1

- Fixed boulders in craters problem
- Tar file does not contain ".svn" or other crud

Server Version 1.1

- Added full-resolution Linux version and Mac OS X version.
- Added `-p` command-line option.
- The server now sets `SO_REUSEADDR`.
- Rotational acceleration is properly modeled.

Server Version 1.2

- Fixed bug in Martian behavior.
- Better messages when the server fails
- Remove bogus `glEnable(GL_DEPTH_TEST)`
- `The server now sets SO_LINGER (except on Mac OS 10.4 versions).`

Back to contest home page