Rainy hills

1. Installation steps

To run application JBoss AS 7.1.1.Final and Java 7 need to be installed. To build project Maven 3 need to be installed.
After do following steps:

a. Unzip project to your local storage.
b. Go to folder with project and in pom.xml file put valid path to JBoss AS 7.1.1.Final folder in tag <jbossHome>.
c. Go to folder with project and in Command Line(Terminal) run command *mvn clean install* to build project and run unit tests.
d. In folder *target* you will find *water-after-rain.war* file.
e. Deploy this WAR file on JBoss server.
f. To test that application is deployed successfully send GET request to URL: http://localhost:8080/water-after-rain/rest/heartbeat. You should get OK(200) response. (It can be doing by using browser or some REST client: for example cURL)
g. For calculating volume you need send POST request to URL: http://localhost:8080/water-after-rain/rest/surface with ContentType: *application/json* and body with JSON like: {"list":[1,2,3,4,5]}. The answer will be in response.

To run integration test, maven profile*: integration-test* need to be enabled. Build project using command:

mvn clean install -P integration-test

During the running integration test JBoss application server will be starting, deployed application, run test requests, undeployed application and shutdown application server.

2. Algorithm description

In first step, we compare borders of the surface. After we define lower border, we start move from lower border to upper border. If we meet unit with less height, than lower border, we fill it with water, else, if we meet unit with greater height, we define this unit as new border and repeat all steps. We finish when left border and right border will be the same unit. During walking, we calculate volume of water that we use to fill units.

Complexity of algorithm is linear (O(n) where n is array size) as for memory as for time, because we don't create new objects and we are passed throw the array once.

3. Application design

The application has layered structure. It consists from web layer, which handles REST request and service layer, which provide calculations. Application build in WAR file and provide two REST endpoints:

- GET - http://<ip>:<port>/water-after-rain/rest/heartbeat;
- POST - http://<ip>:<port>/water-after-rain/rest/surface;

/heartbeat – is for checking that application is alive.

/surface – is for providing calculations.