**public** **void** earliestStart () {

**if** (id == 0) {

//If first activity, earliestStart is 0

earliestStart = 0;

}

**else** {

//Check how many activities the current one depends on

//Then choose the activity with the latest earliestFinish to calculate the earliestStart

earliestStart = Activities[previous].id.earliestFinish;

}

}

**public** **void** earliestFinish () {

//earliestFinish is always earliestStart + duration of the activity

earliestFinish = earliestStart + duration;

}

**public** **void** latestFinish () {

**if** (Activity == last)

//If last activity, latestFinish = earliestFinish of the actvity

latestFinish = earliestFinish;

**else** {

//Check how many activities depend on the current one

//The latestFinish of the current activity is the earliest latestStart of its following activities

latestFinish = nextActivity.latestStart;

}

}

**public** **void** latestStart () {

//latestStart is always latestFinish - duration of the activity

latestStart = latestFinish - duration;

}

**public** **void** activityFloat () {

activityFloat = latestStart - earliestStart;

criticalPath = **true**;

}