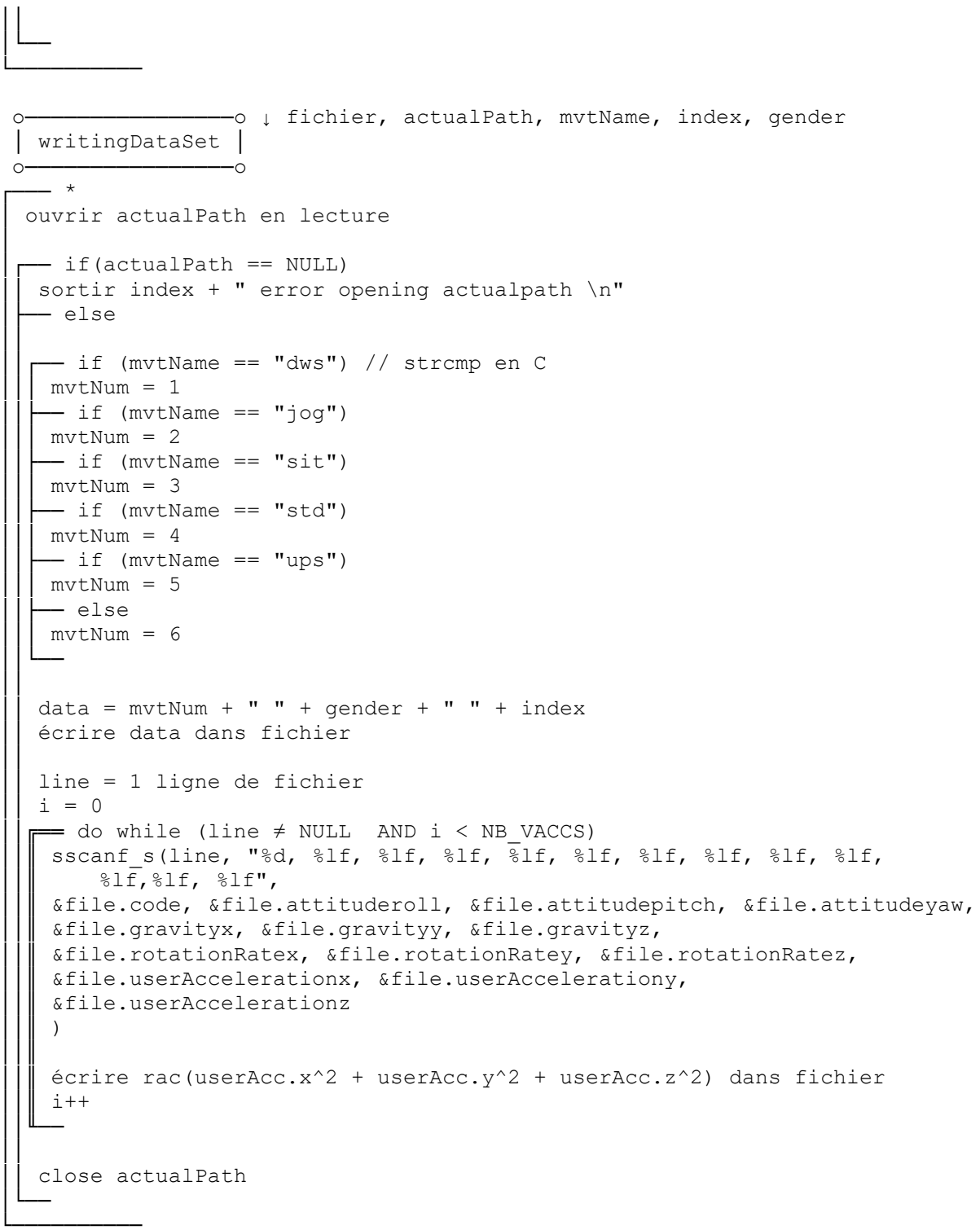
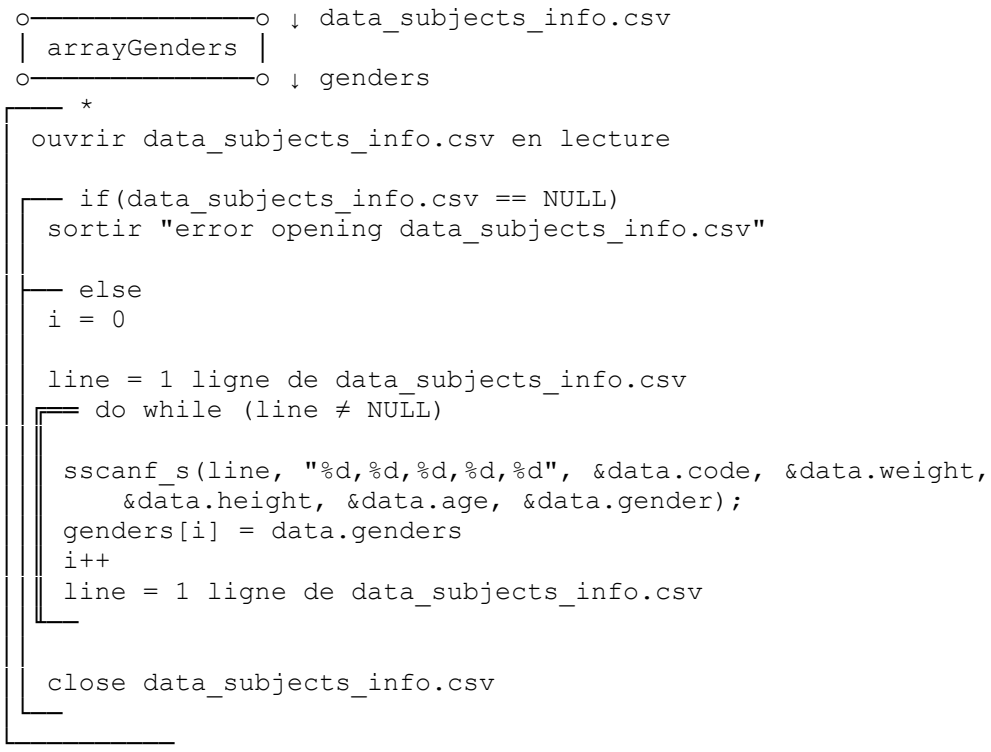
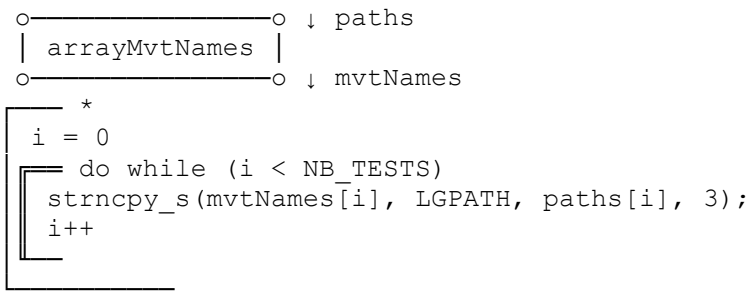
```
// struct data {
//     code
//     weight
//     height
//     age
//     gender
// }

// struct file {
//     code
//     attituderoll
//     attitudepitch
//     attitudeyaw
//     gravityx
//     gravityy
//     gravityz
//     rotationRatex
//     rotationRatey
//     rotationRatez
//     userAccelerationx
//     userAccelerationy
//     userAccelerationz
// }

// LGPATH = 100
// LINE = 1000
// NB_VACCS = 600
// NB_PARTICIPANTS = 24
// NB_TESTS = 15

o------o
| Main |
o------o
---- *
 char paths[][LGPATH] = "dws_1/", "dws_2/", "dws_11/", "jog_9/",
       "jog_16/", "sit_5/", "sit_13/", "std_14/",
 "std_6/", "ups_3/", "ups_4/", "ups_12/", "wlk_7/", "wlk_8/", "wlk_15/"

 subs[][LGPATH] = "sub_1.csv", "sub_2.csv", "sub_3.csv", "sub_4.csv",
       "sub_5.csv", "sub_6.csv", "sub_7.csv",
 "sub_8.csv", "sub_9.csv", "sub_10.csv", "sub_11.csv", "sub_12.csv",
       "sub_13.csv", "sub_14.csv", "sub_15.csv", "sub_16.csv",
 "sub_17.csv", "sub_18.csv","sub_19.csv", "sub_20.csv", "sub_21.csv",
       "sub_22.csv", "sub_23.csv", "sub_24.csv"

 actualPath[LGPATH] = ""

 o----------------o ↓ data_subjects_info.csv
 | arrayGenders |
 o----------------o ↓ genders

 o------------------o ↓ paths
 | arrayMvtNames |
 o------------------o ↓ mvtNames

 // open files trainSet.csv et testSet.csv
 ouvrir trainSet.csv en écriture
 ouvrir testSet.csv en écriture
```

```
if(trainSet.csv == NULL OR testSet.csv == NULL)
sortir "Erreur d'ouverture"
else

écrire "Mouvement, genre, index" dans trainSet

iTrain = 0
do while (iTrain < NB_VACCS)
 écrire "vAcc" + iTrain + 1
 iTrain++


écrire "Mouvement, genre, index" dans testSet

iTest = 0
do while (iTest < NB_VACCS)
 écrire "vAcc" + iTest + 1
 iTest++


iTest1 = 1
iTest2 = 2
indexTest = 1
indexTrain = 1

i = 0
do while (i < NB_TESTS)
 if (iTest2 == NB_PARTICIPANTS)
  iTest1 = 1
  iTest2 = 2


 j = 0
 do while (j < NB_PARTICIPANTS)
  actualPath[LGPATH] = paths[i] + "sub_" + j +".csv"

  if (iTest1 == j OR iTest2 == j)
   o──────────────o ↓ testSet.csv, actualPath, mvtNames[i], indexTest,
                    genders[j]
   | writingDataSet |
   o──────────────o
   indexTest++
  else
   o──────────────o ↓ trainSet.csv, actualPath, mvtNames[i],
                         indexTrain, genders[j]
   | writingDataSet |
   o──────────────o
   indexTrain++


  j++


 iTest1 +=2
 iTest2 +=2
 i++


// close files  trainSet.csv et testSet.csv
close trainSet.csv
close testSet.csv
```


writingDataSet
↓ fichier, actualPath, mvtName, index, gender
*
ouvrir actualPath en lecture
if(actualPath == NULL)
sortir index + " error opening actualpath \n"
else
if (mvtName == "dws") // strcmp en C
mvtNum = 1
if (mvtName == "jog")
mvtNum = 2
if (mvtName == "sit")
mvtNum = 3
if (mvtName == "std")
mvtNum = 4
if (mvtName == "ups")
mvtNum = 5
else
mvtNum = 6
data = mvtNum + " " + gender + " " + index
écrire data dans fichier
line = 1 ligne de fichier
i = 0
do while (line ≠ NULL AND i < NB_VACCS)
sscanf_s(line, "%d, %lf, %lf, %lf, %lf, %lf, %lf, %lf, %lf, %lf,
%lf,%lf, %lf",
&file.code, &file.attituderoll, &file.attitudepitch, &file.attitudeyaw,
&file.gravityx, &file.gravityy, &file.gravityz,
&file.rotationRatex, &file.rotationRatey, &file.rotationRatez,
&file.userAccelerationx, &file.userAccelerationy,
&file.userAccelerationz
)
écrire rac(userAcc.x^2 + userAcc.y^2 + userAcc.z^2) dans fichier
i++
close actualPath


data_subjects_info.csv
arrayGenders
genders
*
ouvrir data_subjects_info.csv en lecture
if(data_subjects_info.csv == NULL)
sortir "error opening data_subjects_info.csv"
else
i = 0
line = 1 ligne de data_subjects_info.csv
do while (line ≠ NULL)
sscanf_s(line, "%d,%d,%d,%d,%d", &data.code, &data.weight,
&data.height, &data.age, &data.gender);
genders[i] = data.genders
i++
line = 1 ligne de data_subjects_info.csv
close data_subjects_info.csv


paths
arrayMvtNames
mvtNames
*
i = 0
do while (i < NB_TESTS)
strncpy_s(mvtNames[i], LGPATH, paths[i], 3);
i++