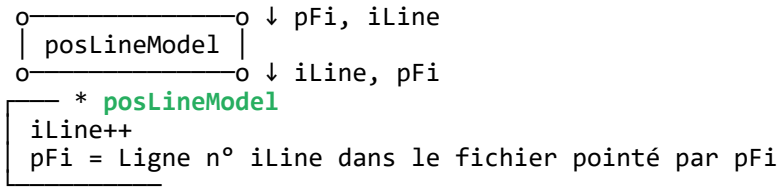
```
struct model{
int mvt;
double v_acc[NB_TEMPS];
};

Constantes :
FI_MODEL = "fiModel.csv"
FI_TEST_SET = "fiTest.csv"
NB_MODELS = 6
NB_DATA = 600
NB_TESTS_MAX = 10000

o————————————————————o
| Evaluation of models |
o————————————————————o
┌── * Evaluation of models
 fopen_s(&pFiModel, FI_MODEL , "r")
 fopen_s(&pFiTest, FI_TEST_SET, "r")

┌── if (pFiModel AND pFiTest)
 o————————————————————o ↓  pFiModel
 | convertFileToTable |
 o————————————————————o ↓  models

 iLine = 1
 pFiTest= se positionner sur la Line n° 1 de "fiTest.csv"

 ┌──────────────────────────────────┐
 | supprimer l'entête de "test.csv" |
 └──────────────────────────────────┘

 nbTests = 0
 ╔══ do while (!eof(pFiTest) AND nbTests < NB_TESTS_MAX)
 ╠ closestDistance = HV

 ╠ o——————————o ↓ pFiTest, iLine
 ╠ | posLine |
 ╠ o——————————o ↓ iLine, pFiTest

 ╠ movement = fscanf_s(pFiTest, "%d", &mov.move)
 ╠ currentMovement = movement


 ╠ o——————————o ↓  line, 4
 ╠ | getV_acc |
 ╠ o——————————o ↓  v_accs, nbV_accs

 ╠ iModel = 0
 ╠═══ do while (iModel < NB_MODELS)

 ╠ o————————————————o ↓  model.v_accs, v_accs, nbV_accs
 ╠ | getDistance |
 ╠ o————————————————o ↓  distance

 ╠ ┌── if (distance < closestDistance)
 ╠ | closestMovement = models[iModel].mvt
 ╠ | closestDistance = distance
 ╠ └──

 ╠ iModel++
 ╠═══

 ╠ estimateClasses[nbTests] = closestMovement
 ╠ realClasses[nbTests] = movement
 ╠ nbTests++
 ╠══


 ╠ o————————————————————————————————o ↓  realClasses, estimateClasses, nbTests
 ╠ | displayResultsForEachClass |
 ╠ o————————————————————————————————o

 ╠ o————————————————————o ↓  realClasses, estimateClass, nbTests
 ╠ | displayAccuracy |
 ╠ o————————————————————o
 ╠
 ╠
```

```
  o──────────────o ↓  realClasses, estimateClass
  │ displayClass │
  o──────────────o ↓  nbTests

  fclose(pFiModel)
  fclose(pFiTest)
├─ else
  sortir "ERREUR : ouverture fichier"
└──
└──────────
```

```
 o──────────────o ↓  model.V_accs, v_accs, nbV_accs
 │ getDistance │
 o──────────────o ↓  distance
┌── * getDistance
 sumDistance = 0
 iV_acc = 0
╔══ do while(iV_acc < nbV_accs)
 sumDistance += (v_accs[iV_acc] - model.V_accs[iV_acc])²
 iV_acc++
╚══

 distance = racine(sumDistance)
└──────────
```

```
 o──────────────────────o ↓  pFiModel
 │ convertFileToTable │
 o──────────────────────o ↓  models
┌── * convertFileToTable
 nbModels = 0

 ┌──────────────────────────────────────┐
 │ supprimer l'entête de "fiModel.csv"  │
 └──────────────────────────────────────┘

╔══ do while (!eof(pFiModel) AND nbModels < NB_MODELS)
 ┌──────────────────────────────┐
 │ lire ligne n° nbModels + 1   │
 └──────────────────────────────┘

 fscanf_s(pFiModel, "%d", &mov.move)

 o──────────o ↓  iLine, 2
 │ getV_acc │
 o──────────o ↓  V_accs, nbV_accs

 model.mvt = movement
 model.v_accs = V_accs
 models[nbModels] = model

 nbModels++
╚══
└──────────
```

```
 o──────────o ↓  iLine, iRow
 │ getV_acc │
 o──────────o ↓  V_accs
┌── * getV_acc
 nbV_accs = 0

 pFiTestSet = ligne n° iLine, colonne n° iRow

╔══ do while (nbV_accs < NB_DATA)
 V_accs[nbV_accs] = valeur pointée par pFiTest;
 nbV_accs++;
 pFiTestSet = ligne n° iLine, colonne n° iRow + nbV_accs
╚══
└──────────
```


posLineModel
↓ pFi, iLine
↓ iLine, pFi
* posLineModel
iLine++
pFi = Ligne n° iLine dans le fichier pointé par pFi