

## Master Informatique - Parcours STL

### UE MU5IN553 - PPC

#### TME 9 - AKKA<sup>1</sup>

#### Carlos Agon

La légende raconte qu'en 1787, alors qu'il venait d'achever la composition de son opéra Don Giovanni et d'une douzaine d'autres œuvres, Mozart prit le temps de mettre au point un petit jeu de salon. Il s'agissait de pouvoir composer de manière aléatoire des menuets au moyen de deux dés et de deux tableaux de 88 nombres, chaque nombre correspondant à une mesure de musique.

##### *Première partie*

	A	B	C	D	E	F	G	H
2	96	22	141	41	105	122	11	30
3	32	6	128	63	146	46	134	81
4	69	95	158	13	153	55	110	24
5	40	17	113	85	161	2	159	100
6	148	74	163	45	80	97	36	107
7	104	157	27	167	154	68	118	91
8	152	60	171	53	99	133	21	127
9	119	84	114	50	140	86	169	94
10	98	142	42	156	75	129	62	123
11	3	87	165	61	135	47	147	33
12	54	130	10	103	28	37	106	5

##### *Seconde partie*

	A	B	C	D	E	F	G	H
2	70	121	26	9	112	49	109	14
3	117	39	126	56	174	18	116	83
4	66	139	15	132	73	58	145	79
5	90	176	7	34	67	160	52	170
6	25	143	64	125	76	136	1	93
7	138	71	150	29	101	162	23	151
8	16	155	57	175	43	168	89	172
9	120	88	48	166	51	115	72	111
10	65	77	19	82	137	38	149	8
11	102	4	31	164	144	59	173	78
12	35	20	108	92	12	124	44	131

Un menuet est une danse ancienne à trois temps (XVIIe siècle). Chaque menuet est divisé en deux parties de huit mesures, soient 16 mesures. Une mesure est un découpage régulier du temps dans une partition. Toutes les mesures ont la même durée (1800 ms). Ainsi, pour exécuter un menuet en entier, il faut en jouer les 16 mesures (8 mesures pour la première partie, 8 mesures pour la seconde).

---

<sup>1</sup> Ce texte est extrait de l'article : Denis Lorrain. « Réalisation de jeux musicaux du XVIIIe siècle : Mozart & Stadler. ». Journées d'Informatique Musicales, Lyon 2003.

## Le jeu des dès de Mozart.

Voici les instructions pour composer autant de menuets que l'on veut par le moyen de deux dés sans avoir la moindre connaissance de la Musique ou de la Composition :

-Les lettres A-H, qui sont placées au-dessus des 8 colonnes des tables de nombres, montrent les 8 mesures de chaque partie du menuet. Par exemple : A, la première, B, la seconde, C, la troisième, etc., et les nombres dans la colonne de sous les lettres montrent le nombre de la mesure.

-Les nombres de 2 jusqu'à 12 montrent la somme des deux dés qu'on peut jeter.

On jette donc par exemple, pour la première mesure de la première partie du menuet, avec 2 dés 6, et cherche près du nombre 6 dans la colonne A, le nombre de la mesure 148 dans la musique. Après cela on jette pour la seconde mesure, par exemple 9. On cherche près de 9 sous B, et on trouve 84 de la table de musique et ainsi cela va à l'infini.

## Travail à réaliser

Vous devez créer un système d'acteurs pour jouer à l'infini des menuets en utilisant le jeu des dés de Mozart.

Commencez pour télécharger le projet *mozartGame\_code* que vous trouverez dans le site du cours. Vous trouverez dans ce projet :

- 1- Un fichier *Main.scala* pour lancer votre application.
- 2- Un fichier *Player.scala* pour jouer une mesure. Vous devez implémenter l'action à réaliser lors de la réception d'un message **case Measure (l)** dans la méthode **receive**.
- 3- Un fichier *DataBase.scala* qui implémente un acteur *DataBaseActor* qui contienne les 176 mesures correspondantes aux deux tables du jeu. Chaque mesure est un **ObjetMusical** défini par les **case classes** suivantes :

```
abstract class ObjetMusical
case class Note (pitch:Int, dur:Int, vol:Int) extends ObjetMusical
case class Chord (date:Int, notes:List[Note]) extends ObjetMusical
case class Measure (chords:List[Chord]) extends ObjetMusical
```

Par rapport au TME 8, nous avons légèrement modifié la hiérarchie de classes pour coder la musique. Il existe toujours une classe abstraite appelée **ObjetMusical**, mais nous utiliseront principalement la classe **Measure** qui contient une liste d'instances de la classe **Chord**. Un **Chord** a une date (l'instant où l'accord commence à jouer) et une liste d'instances de la classe **Note**. Finalement, une note est définie par une hauteur (entier entre 0 et 127, voir le codage midi), une durée (exprimée en millièmes de seconde) et un volume (entier entre 0 et 127).

Vous pouvez envoyer au *DataBaseActor* un message **GetMeasure (num)**. Vous aurez en réponse un message avec une instance de **Measure**.

Une fois le projet chargé vous devez écrire un acteur *Conductor* et un acteur *Provider* et mettre tous les acteurs ensemble pour jouer à l'infini des motets.

*Conductor* reçoit un message **StartGame**, puis lance deux dés et envoie un message **GetMeasure (result)** au *Provider* où **result** est la somme des deux dés. *Provider* enverra un message à *Conductor* avec la mesure qu'il y a obtenue grâce aux deux tables. Il communiquera avec l'acteur *DataBase* pour trouver la bonne mesure.

Une fois que *Conductor* reçoit la mesure, il l'envoie au *Player* et, 1800 ms après, il s'envoie à lui-même le message **StartGame** pour jouer à l'infini.

N'oubliez pas de faire un schéma montrant la façon dont les acteurs communiquent.