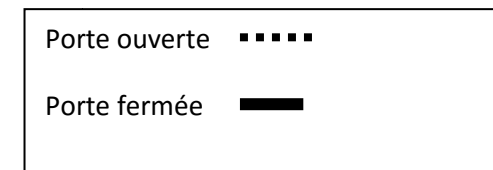
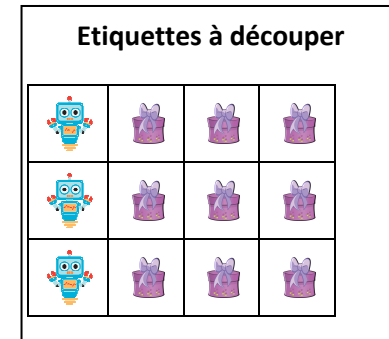
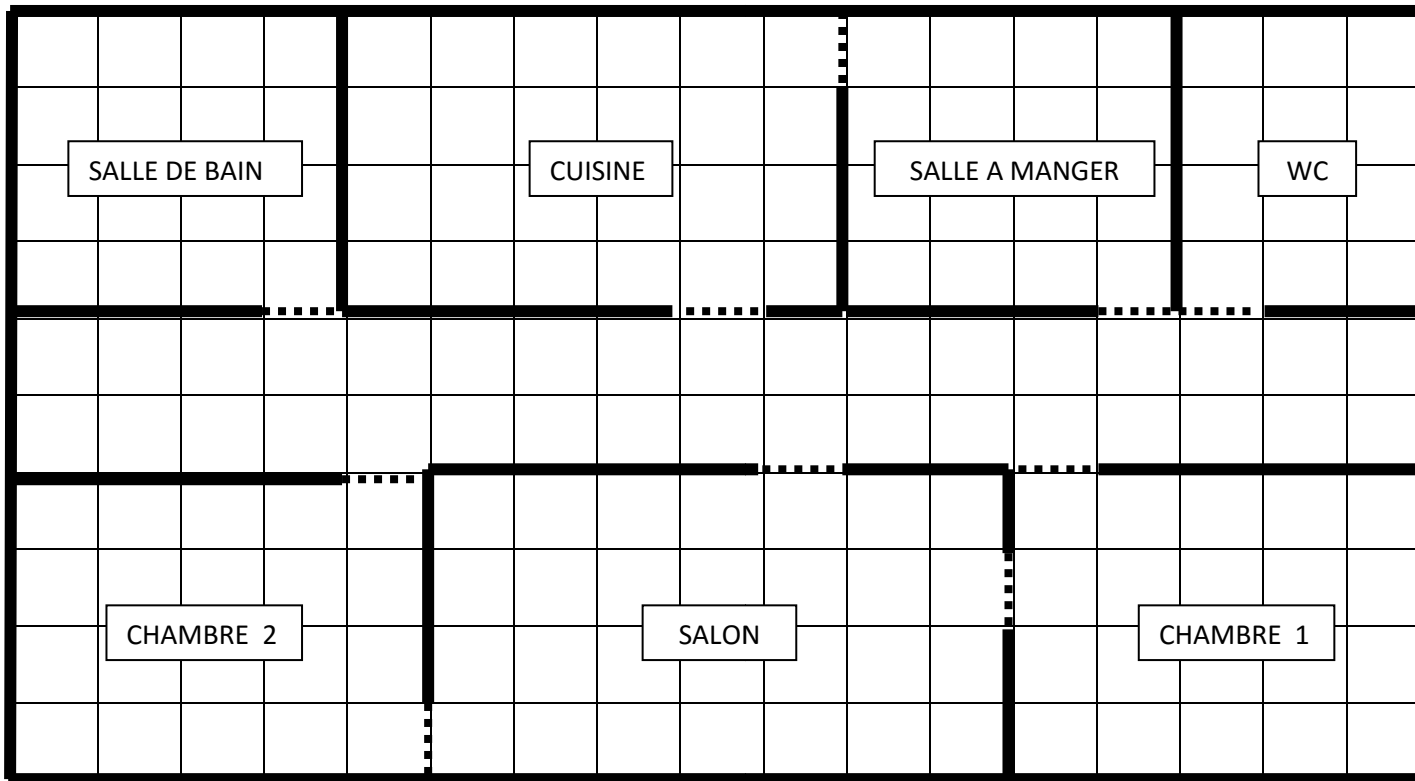


**Domaine : Les langages pour penser et communiquer**

**Sous-domaine : Espace et géométrie.** Initiation à la programmation : « une initiation à la programmation est faite à l'occasion notamment d'activités de repérage ou de déplacement » (Programmes 2016).

- Attendu de fin de cycle : « (Se) repérer et (se) déplacer dans l'espace en utilisant ou en élaborant des représentations »
- Compétence : « Programmer les déplacements d'un robot ou ceux d'un personnage sur un écran »

	Objectifs Langage	Objectifs Mathématiques		
		Phase 1	Phase 2	Phase 3
<b>Se déplacer en fonction de contraintes</b>	<p>Instructions : boucle (répéter X fois), condition (si...alors...sinon)</p> <p>Vocabulaire permettant de définir des positions et des déplacements.</p> <p>Déplacer un personnage</p> <p><b>Variables :</b></p> <p>Nombre d'instructions</p> <p>Nombre d'objets obstacles (portes fermées)</p> <p>Nombre d'espaces de jeu (de pièces)</p> <p>Modalité du travail sur feuille (binôme, individuel)</p>	<p><b>Donner des instructions pour coder un déplacement relatif :</b> l'effet des instructions dépend de l'orientation du personnage (ex : effectue un <math>\frac{1}{4}</math> de tour vers la droite).</p> <p>A opposer au déplacement absolu qui ne dépend pas de l'orientation du personnage (ex : avance de trois pas vers le tableau).</p> <p><u>Consigne</u> : Déplacer le binôme d'une pièce à l'autre en passant par...</p> <p>Par binôme : faire guider le déplacement d'un élève sur une surface quadrillée au sol.</p> <p><b>Lexique :</b> Instruction, avance, recule, tourne à droite, tourne à gauche, s'orienter, initialisation, repère, origine.</p>	<p><b>Utiliser des boucles</b></p> <p>Comment coder le déplacement d'un personnage en utilisant des boucles simples (ex : répéter « avance » cinq fois) ?</p> <p><u>Consigne</u> : sur une feuille avec un quadrillage, utiliser les boucles pour optimiser le nombre d'instructions pour réaliser un parcours donné.</p> <p>Travail sur grille ci-dessous (toutes les portes sont supposées ouvertes)</p> <p><b>Lexique</b> : boucle, répéter</p>	<p><b>Utiliser des conditions</b></p> <p>Comment coder le déplacement d'un personnage en utilisant des conditions ?</p> <p>« Si...alors » (ex : si la porte est fermée alors ouvre la porte)</p> <p><b>OU</b> (plus compliqué)</p> <p>« Si ...alors ...sinon... » (ex : si la porte est fermée alors...sinon avance)</p> <p><u>Consigne</u> : sur une feuille avec un quadrillage, utiliser boucles et/ou condition pour réaliser un parcours donné.</p> <p>Travail sur grille ci-dessous (indiquer au crayon de bois les portes fermées).</p> <p><b>Lexique</b> : condition</p>



**Pour aller plus loin :** Pour favoriser l'initiation à la programmation, l'enseignant pourra proposer à la fois des activités « branchées » (usage d'applications numériques pour apprendre à coder) et des activités « débranchées », comme celles présentées dans ce document.

« [...] l'informatique est une science et une technique faite d'abstractions, et [que] ces abstractions ne sont pas accessibles aux élèves directement. Elles doivent être appréhendées à travers des situations concrètes et de l'expérimentation personnelle. C'est le levier des activités débranchées [...] » (Abiteboul, Archambault, Berry, De la Higuera, Dowek, Nivat, 2013).

Si vous souhaitez investir des activités branchées en continuité de la situation proposée, une première information pourra vous être utile dans le choix de l'application. On peut distinguer deux types d'applications pour apprendre à coder :

- celles qui se focalisent sur le déplacement d'un personnage sur un écran et qui imposent une progression dans l'activité (ex : Run Marco, Lightbot, Box Island, cf. <https://hourofcode.com/fr/learn>),
- celles qui permettent à l'usager de créer lui-même des programmes dans lesquels les personnages pourront effectuer d'autres actions que celle de se déplacer (ex : Scratch Jr, Scratch).