|  |  |
| --- | --- |
| + 3 actions | + 1 PV |
| Echanger tuile | Bouclier éclair |
| Protéger bourrasque | + 1 PV |
| remove(int) | reverse() |
| pop() | shuffle() |
| replace() | + 1 PV |
| countdown()  Implémenter un compte à  rebours allant de 10 à 1  **7 tuiles** | check\_team(List l)  Vérifier que le nombre personnes dans  l’équipage est conforme à la liste donnée.  Il suffit de vérifier la taille de la liste l  **8 tuiles** |
| **Structure de contrôle :**  while  **Variables locales :**  n : valeur du compte à rebours (10,9,8,…)  **Initialisation :**  n = 10  **Primitive :**  show(n) : affiche la valeur de n | **Structures de contrôle :**  if, for  **Variables locales :**  n : valeur du nombre de  membre de l’équipage  **Initialisation :**  n = 0  **Entrée :**  l : liste contenant  les membre de l’équipage |
| check\_fuel(int x,int y)  Vérifier s’il y a assez d’essence  dans le réservoir.  Dans le cas contraire on remplit le réservoir  jusqu’à avoir assez d’essence  pour arriver à destination (et revenir !)  **11 tuiles** |  |
| **Structures de contrôles :**  if, while  **Variable globale :**  k : volume d’essence dans le réservoir (en L)  **Primitives :**  check() : note la tâche comme faite  fill() : rempli le réservoir  **Entrée :**  x : la distance de l’aller (en km)  y : consommation de la fusée (en L/km) |
|  |  |