

TP 2

Structure de données

Shir-Li Kedem

Sciences informatiques

21/03/2023

1 - Checklist d'Avion

1. Pour réaliser ce programme d'aide à la checklist, la structure de données à utiliser est une pile (stack) pour stocker les tâches à effectuer. Une pile est une structure de données qui suit le principe LIFO (Last In, First Out), c'est-à-dire que le dernier élément ajouté est le premier élément à être retiré. Plus précisément, dans le contexte de la checklist, la pile devrait contenir les différentes tâches (tasks) à effectuer, avec les sous-tâches (steps) imbriquées représentées comme des éléments de la pile. Nous avons donc deux structures indispensable pour ce programme :

- La première structure Task qui représente les tâches à effectuer, contient un nom (name), une liste de sous-tâches (steps) ainsi que le nombre de sous-tâches (n_steps).
- La seconde structure est la structure Stack qui représente la pile. Elle contient la liste des tâches relatives à la structure Task (tasks), puis le nombre de tâches (n_tasks).

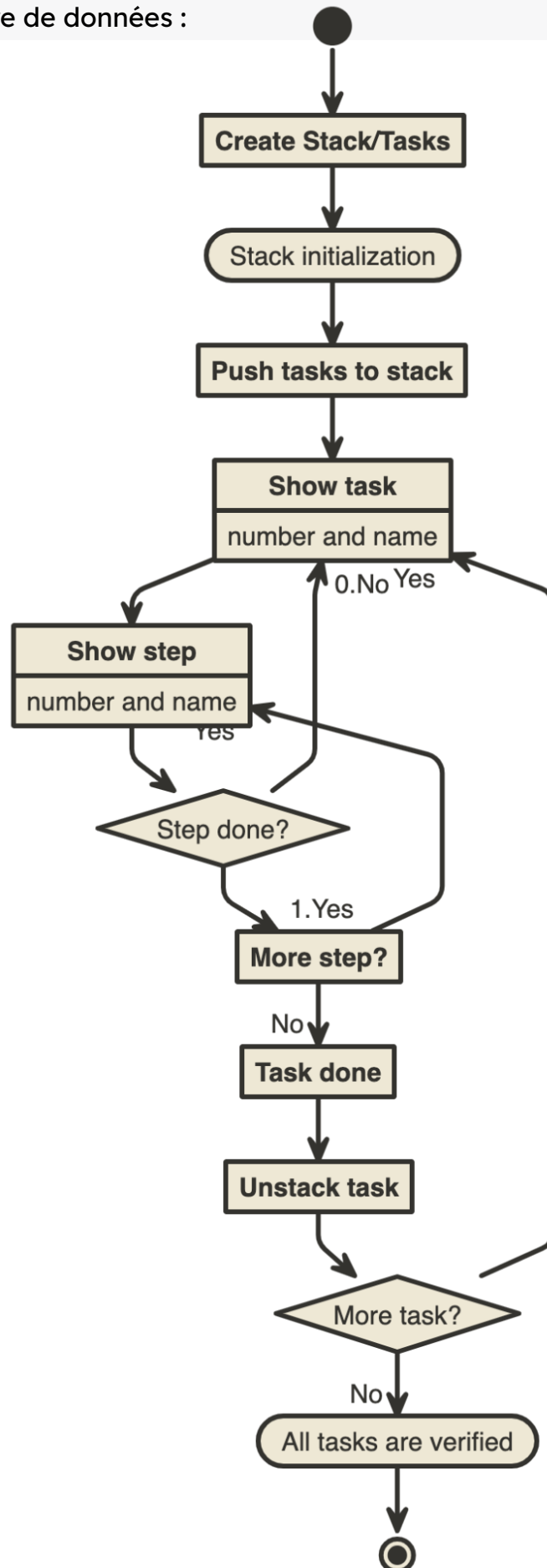
Il y a aussi dans le programme plusieurs fonctions permettant d'interagir avec la pile. Comme par exemple :

- init() : Pour initialiser la pile
- make_task() : Pour initialiser une task
- push() et pop() : Empiler et dépiler une tâche
- print_task() et print_stack() : Afficher la tâche et afficher la pile
- validate_checklist() : Indiquer pour chaque tâche si elle est validée

Lorsqu'une tâche est validée, elle est retirée de la pile, et si une sous-tâche n'est pas validée, toutes les tâches sous-jacentes

(toutes les tâches empilées jusqu'à la tâche de la sous-catégorie correspondante) doivent être retirées de la pile et réévaluées.

2. Le diagramme d'activité complet qui utilise la pile comme structure de données :



2 - Les sièges

1. Dans ce problème, pour simuler la salle d'attente de notre compagnie aérienne où l'affectation des sièges se fait selon le principe du "premier arrivé, premier servi", une structure intermédiaire appropriée serait donc naturellement une file d'attente (qui respecte l'ordre FIFO) pour représenter les passagers qui attendent leur embarquement. Une file d'attente est une structure intermédiaire qui permet d'ajouter des éléments à la fin de la file et de les retirer du début de la file, dans l'ordre d'arrivée.
2. Le diagramme représente un processus de gestion d'une file d'attente de passagers. Au début, une file d'attente vide est créée. Ensuite, des passagers sont ajoutés à la file d'attente avec leur position et leur nom. Un nombre aléatoire de passagers est ensuite retiré de la file d'attente. Sachant que le nombre aléatoire pour cet exercice est entre 1 et 5, cette valeur peut être modifiée pour tout besoin. Si la file d'attente est vide, aucun passager ne peut être retiré. Sinon, un passager est retiré et il est vérifié s'il y en a d'autres à retirer. Une fois que tous les passagers à retirer sont retirés, la file d'attente est affichée. Ensuite, un nombre aléatoire de passagers est ajouté à la file d'attente, et le processus se répète car c'est une boucle de type `while(True)`, c'est-à-dire une boucle infinie.

