

Implantation du menu d'un téléphone portable à l'aide d'automates

1 Description du projet

L'objectif du projet est de réaliser une implantation robuste du mécanisme de navigation entre les différents menus d'une application. L'utilisation d'automates oblige à spécifier l'état de l'application pour chaque événement possible et évite les bugs du « baby player » où l'application entre dans un cas non spécifié lorsqu'un enfant actionne au hasard les éléments de l'interface.

On prend comme exemple le cas d'un téléphone portable mais on pourrait faire le même exercice sur le menu d'un site web ou toute autre interface.

On considère un téléphone avec les fonctionnalités suivantes disponibles sur le clavier :

1. réglage du volume : $\boxed{+}$, $\boxed{-}$
2. lecteur mp3 : \boxed{P} play, \boxed{S} stop
3. appel : $\boxed{0}$, $\boxed{1}$, $\boxed{2}$, $\boxed{3}$, $\boxed{4}$, $\boxed{5}$, $\boxed{6}$, $\boxed{7}$, $\boxed{8}$, $\boxed{9}$ numérotation, \boxed{E} effacer, \boxed{C} composer le numéro (c'est-à-dire appeler)
4. envoi de SMS : \boxed{N} rédiger un nouveau message, $\boxed{0}$, $\boxed{1}$, $\boxed{2}$, $\boxed{3}$, $\boxed{4}$, $\boxed{5}$, $\boxed{6}$, $\boxed{7}$, $\boxed{8}$, $\boxed{9}$ rédaction (pour réduire l'alphabet on ne considère qu'un clavier numérique), \boxed{E} effacer, \boxed{V} envoyer
5. \boxed{A} annuler l'action précédente
6. \boxed{Q} quitter le menu actuel et revenir au précédent
7. prise d'appel (vous ou votre interlocuteur) : \boxed{D} décrocher, \boxed{R} raccrocher
8. réception de SMS : \boxed{L} Lire, \boxed{E} effacer
9. \boxed{O} on, \boxed{F} off

et les événements suivants venus de l'extérieur sont aussi considérés comme des symboles de l'alphabet de l'automate :

1. (#) sonnerie signalant la réception d'un appel
2. (@) sonnerie signalant la réception d'un message
3. (D) l'un des deux correspondants a décroché
4. (R) l'un des deux correspondants a raccroché

Ainsi, $\Sigma = \{ +, -, P, S, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, E, C, N, V, A, Q, D, R, L, O, F, \#, @, L, R \}$

Dans cet exercice on ne s'intéresse pas aux fonctionnalités associées aux événements, ni au temps de réaction (pas de compteurs du temps écoulé) ; seuls nous importe le mode dans lequel se trouve le téléphone et la garantie qu'il ne se retrouve pas dans un mode dont on ne peut pas sortir, ni dans un mode non spécifié.

Les états de l'automate correspondent aux différents modes du téléphone qui comporte au minimum les modes suivants :

1. off,
2. accueil,
3. mp3,
4. numérotation,
5. appel,
6. rédaction sms,
7. envoi sms,
8. sonnerie tél,
9. sonnerie sms,
10. communication,
11. lecture sms,
12. ...

Votre implantation devra prendre en entrée une chaîne de caractères constituées de symboles de l'alphabet de votre automate. Cette chaîne correspond aux événements (à la fois ceux du clavier et les événements extérieurs). Votre interface devra se comporter de la manière suivante (indéfiniment) :

1. afficher le mode/état courant (le nom du mode/état doit être explicite)
2. lire le caractère courant de la chaîne d'événement
3. déterminer le mode suivant

Exemple (version séquence d'événements)

```
séquence d'événements ? ['0' ; 'F']
-> off
0 -> accueil
F -> off
```

Pour tester votre automate il est plus simple de faire une version qui lit des événements/caractères entrés un par un au clavier ; il ne sera pas difficile de l'étendre pour obtenir une implantation qui prend en entrée une chaîne de caractère et qui est plus pratique pour une démo.

Exemple (version événement un par un)

```
-> off
? 0
-> accueil
? F
-> off
```

2 Extension (lorsque nous l'aurons vu en cours)

Dans un second temps vous prendrez en compte la prise d'appel ou de message en cascade avec retour au mode/état précédent lors de la fin de la prise de l'appel ou de la fin de la lecture du message.

Pour implanter la mémorisation de l'état de retour vous ajouterez à votre automate une pile d'état (c'est-à-dire un tableau qui stocke les états de retour).

Certaines transitions de l'automate devront placer l'état courant sur le sommet de la pile.

D'autres transitions devront prendre l'état en sommet de pile comme état cible.

3 Conseil

Adoptez les mêmes conventions d'alphabet et d'état que vos collègues ainsi vous pourrez réutiliser leurs jeux de tests et comparer le comportement de vos « téléphones » respectifs.

4 À rendre

- Un document de quelques pages décrivant les spécificités de votre téléphone, les modes et les symboles utilisés
- une implantation en C
- un jeu de tests qui permet d'atteindre chacun des modes et de revenir au mode d'accueil.