



On définit les deux struct suivantes:

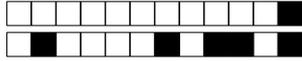
```
typedef struct _cell
{
    char *mot;
    struct _cell *next;
} cell_t;
```

```
typedef struct _liste
{
    cell_t *premier;
    cell_t *dernier;
} liste_t;
```

c) (3 points) Écrivez une fonction qui reçoit une liste et un pointeur vers une chaîne de caractères contenant un seul mot. Cette fonction rajoute une nouvelle cellule ayant pour contenu le pointeur un\_mot à la fin de la liste chaînée liste. Cette fonction doit mettre à jour les champs premier et dernier. Pour une liste vide ces champs auront la valeur NULL.

```
void enfiler_mot(liste_t *liste, char* un_mot) {
```

```
}
```



d) (3 points) Utilisez les trois fonctions ci-dessus pour implémenter une fonction qui prend en paramètre une chaîne de caractères `texte` et retourne la liste de mots qui s'y trouvent dans l'ordre de leur apparition.

```
liste_t* créer_liste(const char *texte) {
```

```
}
```

e) (2 points) Implémentez une fonction qui libère la mémoire d'une liste.

```
void libérer(liste_t *pliste) {
```

```
}
```