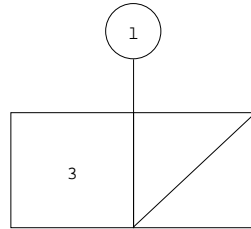


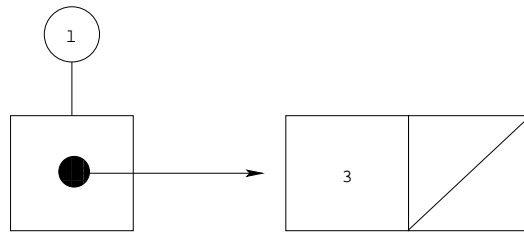
En Java, le contenu du champ `tl` peut être ou bien `null`, ou bien une référence. En C, c'est toujours une référence, mais il y a une référence spéciale `NULL` qui ne peut jamais être associée dans la mémoire et qui joue donc le même rôle que `null` en Java. Cependant, cette construction `NULL` ne fait pas partie du mécanisme des enregistrements, mais du mécanisme des références, comme les constructions `&`, `*`, ... On peut donc construire une liste singleton ainsi

```
struct List l = {3,NULL};
```

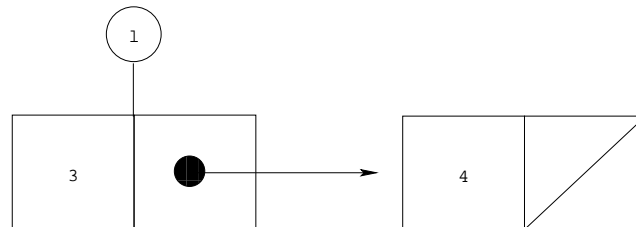
Remarquons que l'état construit en C est



et non comme en Java



Pour construire une liste à deux éléments



il est nécessaire d'être capable de créer une cellule. Pour cela, il y a une construction homologue au `new` de Java, mais qui est, encore une fois, indépendante du mécanisme des enregistrements.

Pour allouer une cellule, en C, on utilise la fonction `malloc`. Et on construit la liste ci-dessus ainsi

```
struct List l;  
l.hd = 3;
```