

```
1 #define MAX 100
2 extern void initFile(void);
3 extern void enfiler(int v);
4 extern int defiler(void);
5 extern int vide(void);
6
7 #include "file.h"
8 static int file[MAX + 1], debut = 0, fin = 0;
9 extern void initFile(void) {
10     debut = fin = 0;
11 }
12 extern void enfiler(int v) {
13     file[fin++] = v;
14     if(fin > MAX) fin = 0;
15 }
16 extern int defiler(void) {
17     int v = file[debut++];
18     if(debut > MAX) debut = 0;
19     return v;
20 }
21 extern int vide(void) {
22     return debut == fin;
23 }
```

FIGURE 1 – **File simple.**

En cours :

- Consulter la documentation sur GraphViz puis Postscript sur la page :  
<http://expreg.org/amsi/C/OD0405S1/>
- Implémenter et visualiser à l'aide de GraphViz un arbre binaire d'entiers.

Exercices :

- Utiliser la file pour implémenter un algorithme de recherche, en largeur d'abord, du plus court chemin dans un labyrinthe (Dijkstra avec toutes les pondérations égales à 1) ;
- Postscript : Dessiner le labyrinthe en utilisant des carrés en guise de murs ;
- Postscript : Dessiner le labyrinthe en utilisant des lignes ;
- Postscript : Calculer et dessiner le plus court chemin (d'un point A à un point B) dans le labyrinthe.