

TD 3

Allocation dynamique

Programmation C
Semestre 2

1

```
1  /* declarations */
2  #include <stdlib.h>
3  #include <stdio.h>
4
5  #define LG_MAX 30
6  typedef int * PtrEntier;
7
8  PtrEntier tabVal;
9
10 /* allocation memoire dynamique du tableau d'entiers */
11 tabVal=(PtrEntier)malloc(LG_MAX * sizeof(int));
12
13 if(tabVal == NULL)
14 {
15     /* afficher un message d'erreur et arreter le processus */
16     perror("echec malloc");
17     exit(1);
18     /* ou lever une exception (sera vu au TD4) */
19 }
20
21 /* exemple d'utilisation du tableau */
22 tabVal[i]= -25;
```

2

```
1  /* declarations */
2  #include <stdlib.h>
3  #include <stdio.h>
4  #define LG_MAX 256
5
6  typedef char * PtrCar;
7
8  PtrCar tampon;
9
10 /* allocation memoire dynamique du tableau de caracteres */
11 tampon=(PtrCar)malloc(LG_MAX+1);
12
13 if(tampon == NULL)
14 {
15     /* afficher un message d'erreur et arreter le processus */
```

```
16     perror("echec malloc");
17     exit(1);
18     /* ou lever une exception (sera vu au TD4) */
19 }
20
21 /* exemple d'utilisation du tableau */
22 /* saisie d'une chaine de caracteres au clavier */
23 fgets(tampon, LG_MAX+1, stdin);
```

3

```
1  /* declarations */
2  #include <stdlib.h>
3  #include <stdio.h>
4  #include <string.h>
5
6  typedef char TabCar60 [60+1];
7
8  typedef struct
9  {
10     int codeProd;
11     TabCar60 designation;
12     float puHT;
13 } EnregProd;
14
15 typedef EnregProd * PtrEnregProd;
16
17 PtrEnregProd unProduit;
18
19 /* allocation memoire dynamique de l'enregistrement unProduit */
20 unProduit=(PtrEnregProd)malloc(sizeof(EnregProd));
21
22 if(unProduit == NULL)
23 {
24     /* afficher un message d'erreur et arreter le processus */
25     perror("echec malloc");exit(1);
26     /* ou lever une exception (sera vu au TD4) */
27 }
28
29 /* exemples d'utilisation de l'enregistrement unProduit */
30 unProduit->codeProd=12345;
31 (* unProduit).puHT=20.45;
32 strcpy(unProduit->designation, "chocolat poudre YABON 250g");
```