

# Programmation avancée

# Allocation Dynamique

Walter Rudametkin

Walter.Rudametkin@polytech-lille.fr  
<https://rudametw.github.io/teaching/>

Bureau F011  
Polytech'Lille

CM3

# Allocation de mémoire

## Variables automatiques

- ▶ Variables de bloc, paramètres de fonctions
- ▶ Créées automatiquement à l'exécution
- ▶ Allocation dynamique sur la pile (stack)

## Variables dynamiques

- ▶ Créées et détruites dynamiquement et explicitement
- ▶ Fonctions `malloc` et `free`
- ▶ Allocation sur le tas (heap)

# Erreur d'allocation

*/\* À ne pas faire \*/*

```
int * allouer_entier() {  
    int var_static ; // alloué sur la pile  
    printf("var_static address is : %p\n",  
           &var_static);  
    return &var_static ;  
    /* var_static est libéré lors  
    de la fin de la fonction */  
}
```

# Allocation dynamique — malloc

## Fonction malloc

- ▶ `void * malloc (size_t taille);`
  - ▶ Alloue dynamiquement dans le tas un espace de `taille` octets
  - ▶ Résultat : *pointeur non typé* vers la zone allouée
  - ▶ Pointeur peut être converti automatiquement vers le type désiré (conversion implicite)
  - ▶ Besoin de `#include<stdlib.h>`

# Allocation dynamique — Exemples

## Allocation dynamique d'un entier

```
int *pt;  
//pt = (int *) malloc(sizeof(int));  
pt = malloc(sizeof(int));  
*pt = 42; //utilisation
```

## Allocation dynamique d'un tableau d'entiers

```
int n; int *pt;  
scanf("%d", &n);  
pt = (int *) malloc(n*sizeof(int));  
//Accès  
*pt = 11; //premier élément  
*(pt+1) = 22 ; //deuxième  
pt[2] = 33 ; //troisième  
*(pt+n-1) = 9876 ; //dernier
```

# Allocation dynamique — Structures

# Allocation dynamique — Structures

Tableau de structures

# Allocation dynamique — Liste chaînée



# Allocation dynamique — Liste contiguë

# Fonction free

- ▶ `void free(void *ptr);`
  - ▶ libère l'espace mémoire pointé par `ptr` (précédemment alloué)
- ▶ Exemple d'utilisation:  
*Suppression du dernier élément de la liste*

```
free(l.espace[l.dernier]);  
l.dernier -= 1;
```

# Listes chaînées — Implantation en C

---

# Listes chaînées — Implantation en C

---

+

# Listes chaînées — Recherche d'un élément

# Listes chaînées — Exemple: ajout en tête