

Chapitre 2. Les pointeurs

Les pointeurs et chaînes de caractères

Une chaîne de caractère est un tableau

Un tableau est un pointeur

Donc une chaîne est un pointeur

`void estMaj(char ch[])`



`void estmaj (char *ch)`

```

main()
{
char nom[20]="ali",*s;
s=nom;
printf("%s",s)
s="bonjour";
printf("%s",s)
}

```

$\&nom[0] = nom = 00FF1EA0$

→ ali

$*s = 'a'$

$*s = *(s + 0)$
 $= s[0] = nom[0]$

00FF1EA0

00FF1EA1

00FF1EA2

00FF1EA3

'a'
'l'
'i'
'\0'
.
.
.
00
FF
1E
A0

nom[0]

nom[1]

nom[2]

s

00FF2B00

```
int t[4]={10,20,30,40} ;
int * ad[4] ;
for(i=0 ;i<4 ;i++)
    ad[i]=t+i
```

```
for(i=0 ;i<4 ;i++)
    printf("%d" , *ad[i])
```

$\longrightarrow 10\ 20\ 30\ 40$

```
printf("%d %d \n",*(ad[1]+1),*(ad[1]+1))
```



$t[2]$

$30\ 20$



$t[1]+1$

$t+i = \&t[i]$

# 0	10	t[0]
# 1	20	t[1]
# 2	30	t[2]
# 3	40	t[3]
	.	
	.	
	.	
# 0	# 1	ad[0]
# 1	# 2	ad[1]
# 2	# 3	ad[2]
# 3	# 4	ad[3]

Exercice 4

Ecrire une fonction `void affiche_entiers(int* tab,int nb)` qui prend en argument un tableau d'entier et le nombre d'éléments qu'il contient et qui affiche ces éléments. N'utilisez pas la notation `tab[i]` pour parcourir le tableau, mais plutôt l'opérateur `++`.

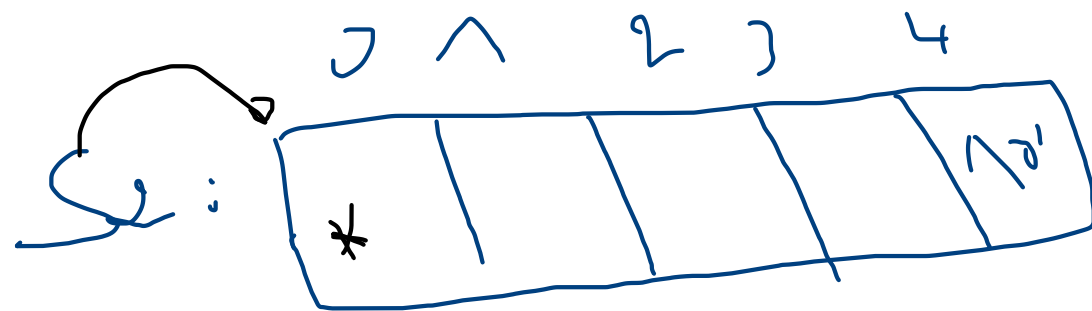
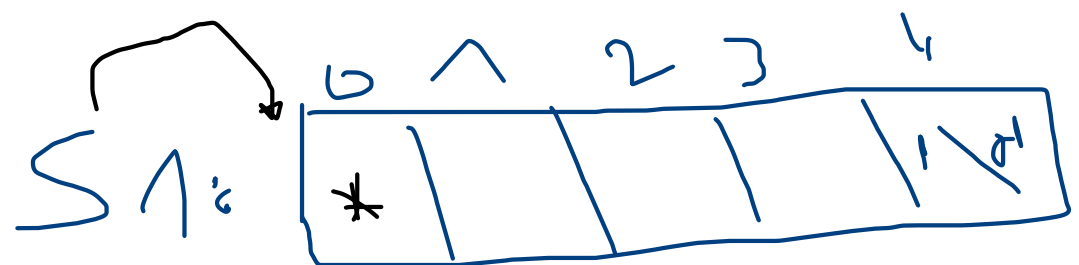
```
void  afficher_entiers(int *tab,int nb)
{
int i;
for(i=0;i<nb;i++)
    printf("%d",tab[i]);  —————>  printf("%d",*(tab+i))
}
```

```
void afficher_entiers_S3(int *tab,int nb)
{
int *p;
for(p=tab;p<tab+nb;p++)
    printf("%d\t",*p);
}
```

```

while (*s1 != '\0' && *s1 == *s2)
{
s1++;
s2++;
if (*s1 < *s2)
return (-1);
else
return (.....);
}

```



s_1

Exercice 1

Ecrire de deux façons différentes, un programme qui lit 10 nombres entiers dans un tableau avant d'en chercher le plus grand et le plus petit

Q1. En utilisant uniquement le formalisme tableau

Q2. En utilisant le formalisme pointeur à chaque fois que cela est possible

Exercice 2

Q1. Ecrire une fonction d'entête `float rand01(void)` renvoie un réel aléatoire dans l'intervalle $[0, 1[$. Utiliser la fonction `rand()` de la bibliothèque `stdlib`.

Q2. Ecrire une fonction d'entête `int myRand(int n)` renvoie un entier aléatoire dans l'intervalle $[0, n[$.

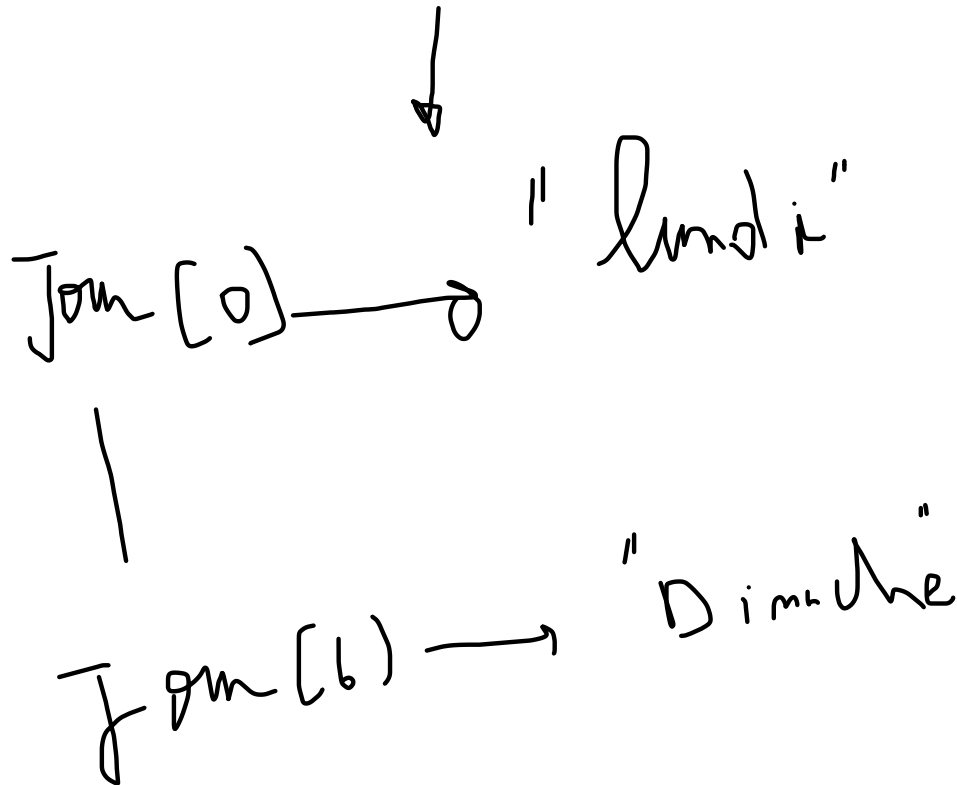
Q3. Ecrire une fonction d'entête `void printTab(int *tab, int lg)` affiche un tableau d'entiers. Est-il possible d'écrire une fonction d'affichage générique capable d'afficher n'importe quel tableau et pas seulement un tableau d'entiers ?


Q4. Ecrire une fonction d'entête `void printMat(int **mat, int n, int m)` affiche une matrice $n \times m$ d'entiers.

Exercice 4

Ecrire un programme qui demande à l'utilisateur de lui fournir un nombre entier entre 1 et 7 et qui affiche le nom du jour de la semaine ayant le numéro indiqué.

0 1 2 3 4 5 6
char jour[7][10]={"Lundi","Mardi","Mercredi","jeudi","Vendredi","Samedi","Dimanche"};




char *jour[7]={"Lundi","Mardi","Mercredi","jeudi","Vendredi","Samedi","Dimanche"};

jour[0] = "L"

|
|

jour[1] = "D"

jour(0)

#10	#1
	#2
	#3
	#4
	#5
	#6
	#7
	#8
	#9
	#10
	#11
	#12
	#13
	#14
	#15
	#16
	#17
	#18
	#19
	#20
	#21
	#22
	#23
	#24
	#25
	#26
	#27
	#28
	#29
	#30
	#31
	#32
	#33
	#34
	#35
	#36
	#37
	#38
	#39
	#40
	#41
	#42
	#43
	#44
	#45
	#46
	#47
	#48
	#49
	#50
	#51
	#52
	#53
	#54
	#55
	#56
	#57
	#58
	#59
	#60
	#61
	#62
	#63
	#64
	#65
	#66
	#67
	#68
	#69
	#70
	#71
	#72
	#73
	#74
	#75
	#76
	#77
	#78
	#79
	#80
	#81
	#82
	#83
	#84
	#85
	#86
	#87
	#88
	#89
	#90
	#91
	#92
	#93
	#94
	#95
	#96
	#97
	#98
	#99

jour(6)

Lundi

#10

Exercice 3

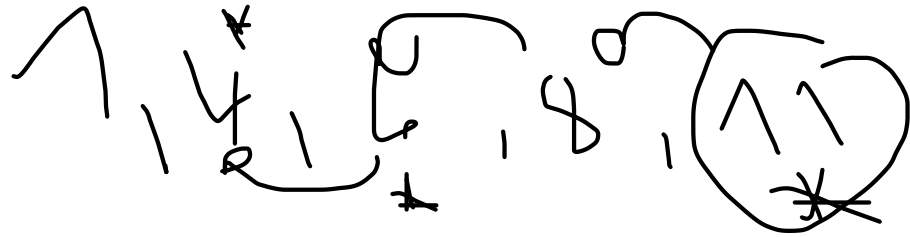
Ecrire un programme qui lit un entier X et un tableau A du type int au clavier et élimine toutes les occurrences de X dans A . Le programme utilisera les pointeurs P1 et P2 pour parcourir le tableau.

Exemple : X=12

A[]={15,12,12,11,8,10,12,11,5,12,6,12,18,20,12,16}

Résultat : A[]={15,11,8,10,11,5,6,18,20,16}

Comment supprimer un element x d'un tableau



$$T[i] = T[i+1]$$

$$x = 4$$

$$\text{if } (T[i] == x)$$

$$\text{for } (j = i, j < N-1, j++)$$

$$T[j] = T[j+1]$$



$$\underline{x = 4}$$

$$j = 1$$

$$T[1] = T[2]$$

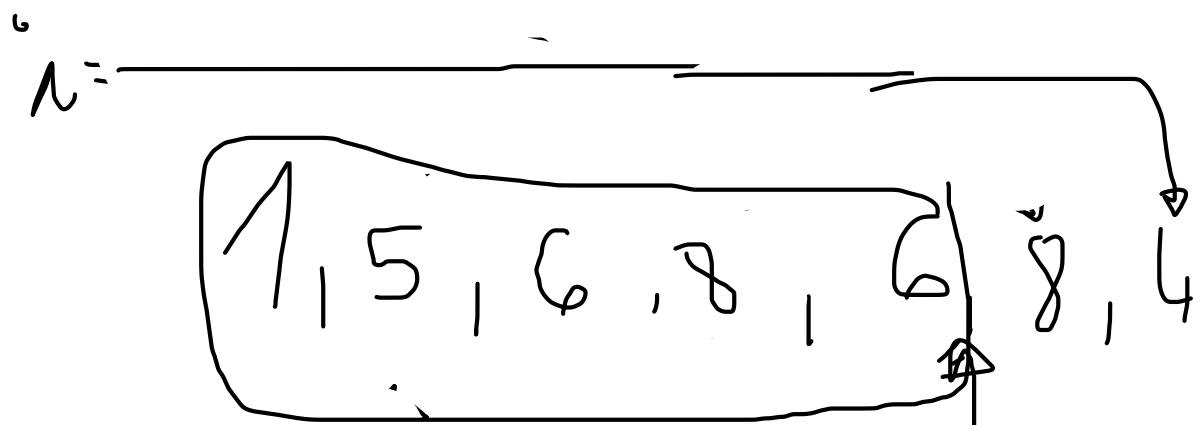
$$T[2] = T[3]$$

$$T[3] = T[4]$$

```

if (T[i] == x)
  for(j=i; j<5; j++)
    T[j] = T[j+1];

```



$$k = 4$$

$$T(0) = 1 \neq 4$$

Diagram illustrating a loop condition and pointer update:

```

graph LR
    k[k=4] --> Cond["if (T[i] != n)"]
    Cond --> Body["{ T[k] = T[i]; k++; }"]
  
```

The diagram shows a loop structure. The initial value of k is 4. The loop condition is $\text{if } (T[i] \neq n)$. If the condition is true, the loop body executes: $T[k] = T[i]$ and $k++$.

Exercice 5

Ecrire un programme qui lit deux tableaux d'entiers A et B et leurs dimensions N et M au clavier et qui ajoute les éléments de B à la fin de A. Utiliser deux pointeurs PA et PB pour le transfer et afficher le tableau résultant A.

$$A = \{1, 3, 4\}$$

$$B = \{2, 6\}$$

→

$$A = \{1, 3, 4, 2, 6\}$$

$$N = 100$$

$$M = 20$$

