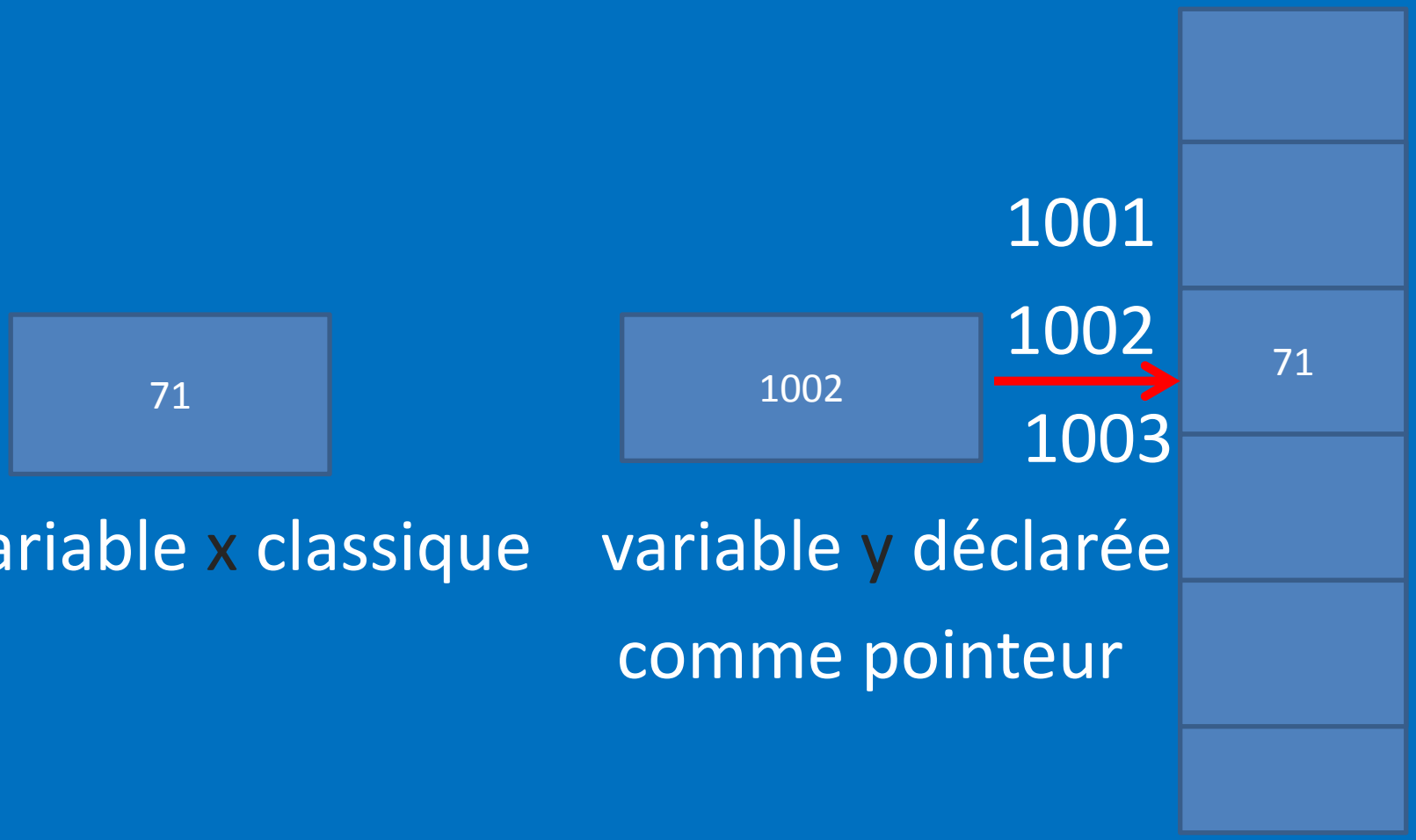
Les pointeurs

Lorsqu'on déclare une variable et ce, quel que soit le langage de , le compilateur réserve en mémoire l'espace nécessaire au contenu de cette variable à une adresse donnée. Toute variable possède donc une adresse en mémoire. La plupart du temps on ne s' intrésse pas à ces adresses. Mais quelque fois, ce type de renseignement peut s'avérer fort utile.

Un pointeur est une variable qui au lieu de contenir l'information proprement dite, contient son adresse en mémoire.

Les pointeurs



Les pointeurs

A la différence de la variable x classique qui contient directement l'information « 71 », la variable y déclarée comme pointeur contient son adresse. On dit que y pointe vers l'information « 71 » grâce à son contenu, qui est l'adresse de cette information.

La syntaxe de déclaration d'un pointeur est la suivante:

Type < nom du type > = ^ type;

Variable < nom de la variable > : <nom du type>

Les pointeurs

Exemple :

Type ptr_entier = ^entier ;

Variable ptr : ptr_entier ;

La variable ptr est appelée à contenir l'adresse d'un entier.

La fonction Allouer (ou New()) permet d'allouer un espace mémoire et fournit *l'adresse de cet espace* en retour.

Cette adresse est assignée au pointeur de même type spécifié entre parenthèse.

La fonction libérer (ou dispose()) permet de libérer un emplacement préalablement alloué lorsqu'on en a besoin.

Les pointeurs

Allouer et libérer des espaces mémoires au besoin s'appelle :

gestion dynamique de la mémoire.

Cette façon de faire est utilisée dans les structures avancées tel que les listes chainées, les piles, les files, etc.

Elle est également utilisée dans le renvoi (le retour) d'enregistrements par adresse par fonction.

Les pointeurs

Renvoi d'un enregistrement par adresse par fonction:

Supposons qu'on souhaite programmer une fonction qui renvoie un enregistrement par adresse. Sachant qu'à la fin de l'exécution d'une fonction, toutes les variables locales de cette fonction sont détruites.

Pour éviter cette destruction avant que le programme principal (ou la fonction ou procédure appelante) ait récupéré les informations contenus dans l'enregistrement renvoyé par la fonction, on effectue une allocation de mémoire avec la fonction « Allouer() ».

Cet espace mémoire ne sera par conséquent pas « détruit » « libéré » jusqu'à ce qu'on l'autorise avec la fonction « dispose () ou bien liberer()».

Les pointeurs

Exemple :

Ecrire un programme qui contient une fonction saisie() qui effectue la saisie d'un enregistrement, puis le renvoie par adresse au programme principal qui l'affiche.

Les pointeurs

```
Programme pointeur;
//définition du type pointeur

Type pointeur = ^vehicule ;
        vehicule = enregistrement
                        no_serie, marque : chaine [ 20 ];
                        annee : entier
        fin;
```

Les pointeurs

```
// définition de la fonction saisie
Fonction saisie() : pointeur;
Variable x : pointeur;
Debut
        Allouer(x);
        ecrire (‘ donnez le numéro de série et marque’);
        lire(x^.no_serie, x^.marque);
        Ecrire(‘donnez l’’année : ‘);
        lire (x^.année);
        saisie := x
Fin;
Variable ptr : pointeur;
```

Les pointeurs

```
Debut
        ptr := saisie() ;
        ecrire(‘ information : ‘, ptr^.no_serie , ‘  ‘ ,
                ptr^.marque , ‘ ‘ , ptr^.annee) ;
        liberer (ptr)
Fin.
```

Les pointeurs

le passage et le retour de paramètres par adresse permet de réduire le temps d'exécution d'un programme ainsi que la mémoire nécessaire à son exécution, car au lieu de manipuler la variable elle-même, on manipule juste son adresse.