Support de cours

Cours:

Initiation à la programmation (en C++)

Vidéo:

Passage des arguments (partie 3)

Concepts (extraits des sous-titres générés automatiquement) :

Fonction capable. Paramètre local de la fonction. Première façon. But de la fonction. Seule façon possible. Fonction de saisie. Coordonnées cartésiennes. Profit du mécanisme du passage. Résultat de la saisie. Fonction saisie. Fonction. Endroit du programme. Exemple du programme principal. Zone mémoire. Seul résultat.

vers la recherche de séquences vidéo
(dans Initiation à la programmation (en C++).)

vers la vidéo

EPFL

Fonctions : passage des arguments
(Partie 3)

Initiation à la programmation (C++)

Vincent Lepetit, Jean-Cédric Chappelier et Jamila Sam

...

notes

résumé

0m 0s

Utilisation du passage par référence

EPFL

Quand utiliser un passage par référence ?

☞ lorsque l'on souhaite modifier une variable

Par exemple :

- pour saisir une valeur :

```
void saisie_entier(int& a_lire);
...
int i(0);
...
saisie_entier(i);
```

Alternative : retourner la valeur :

```
int saisie_entier();
...
i = saisie_entier();
```

- pour « retourner » plusieurs valeurs :

```
void cartesiennes_vers_polaires(double x, double y,
                                double& angle, double& rayon);
```

Alternative : utiliser les structures (futur cours)

- pour « échanger » des variables :

```
void swap(int& i, int& j);
```

Donc concrètement, dans quelle situation va-t-il être utile de recourir au passage par référence ? Et bien, simplement lorsque l'on souhaite qu'une fonction soit capable de modifier une variable qui lui est passée en argument. Supposons par exemple que je souhaite écrire une fonction capable de saisir un entier. Donc le but de la fonction serait de demander à l'utilisateur de saisir au clavier un entier, et bien évidemment nous souhaitons récupérer la valeur saisie afin de pouvoir l'exploiter dans un autre endroit du programme. Comment faire pour récupérer cette valeur saisie ? Une première façon de procéder consiste effectivement à utiliser le passage par référence. Donc dans le programme qui appelle la fonction de saisie, on prévoit une variable qui va stocker le résultat de la saisie, et on passe cette variable à la fonction saisie. En fait lorsque la fonction s'exécute, l'utilisateur va introduire un nombre, va saisir un nombre, et ce nombre va être saisi dans le paramètre local de la fonction, supposant que l'utilisateur saisisse un 2 alors comme nous avons vu que lors du passage par référence, l'argument passé est exactement la même zone mémoire que le paramètre, si on a saisi un 2 dans « a_lire » il va se trouver également dans « i », ce qui veut dire que lorsque la fonction a terminé de s'exécuter, « i » se trouve avec la valeur saisie 2. Evidemment, il ne s'agit pas de la seule façon possible de pouvoir programmer cette fonction de saisie. La façon la plus naturelle et sans doute la meilleure est celle qui va utiliser la valeur de retour pour récupérer le résultat de la saisie. Donc ici notre fonction saisie de trouverait sans argument Lorsqu'elle s'exécute, elle va demander une valeur à l'utilisateur qui va la saisir dans une variable locale à la fonction puis la retourner pour qu'elle puisse être exploitée par le monde extérieur. Donc ici,

notes

résumé

0m 1s

Utilisation du passage par référence

Quand utiliser un passage par référence ?

☞ lorsque l'on souhaite modifier une variable

Par exemple :

- pour saisir une valeur :

```
void saisie_entier(int& a_lire);
...
int i(0);
...
saisie_entier(i);
```

Alternative : retourner la valeur :

```
int saisie_entier();
...
i = saisie_entier();
```

- pour « retourner » plusieurs valeurs :

```
void cartesiennes_vers_polaires(double x, double y,
                                double& angle, double& rayon);
```

Alternative : utiliser les structures (futur cours)

- pour « échanger » des variables :

```
void swap(int& i, int& j);
```

on serait dans la nécessité de recourir à l'affectation pour récupérer la valeur de retour de la fonction de saisie dans une variable, par exemple du programme principal qui sera utilisée dans le programme principal.

notes

résumé

Utilisation du passage par référence

EPFL

Quand utiliser un passage par référence ?

☞ lorsque l'on souhaite modifier une variable

Par exemple :

- pour saisir une valeur :

```
void saisie_entier(int& a_lire);
...
int i(0);
...
saisie_entier(i);
```

Alternative : retourner la valeur :

```
int saisie_entier();
...
i = saisie_entier();
```

- pour « retourner » plusieurs valeurs :

```
void cartesiennes_vers_polaires(double x, double y,
                                double& angle, double& rayon);
```

Alternative : utiliser les structures (futur cours)

- pour « échanger » des variables :

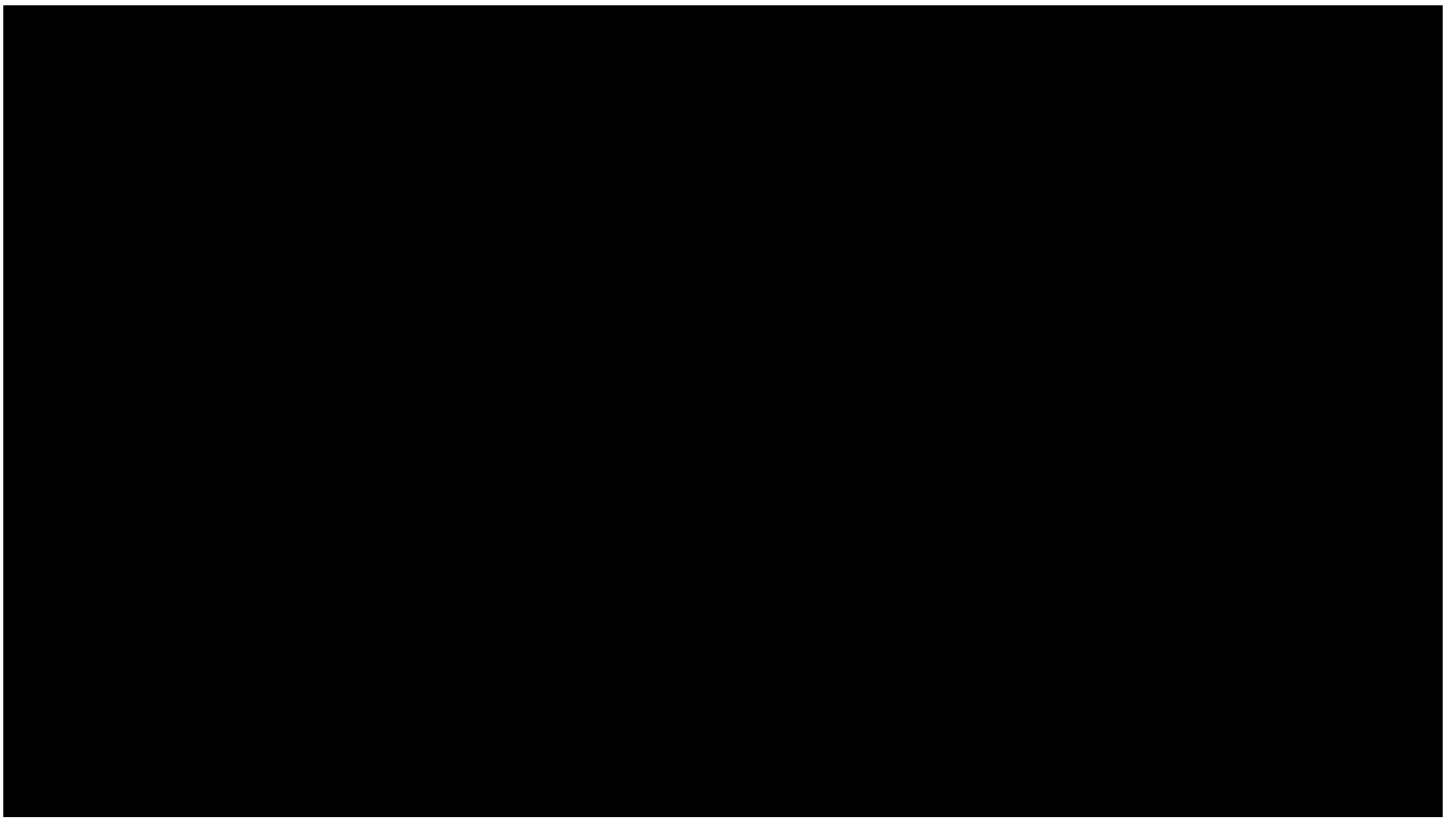
```
void swap(int& i, int& j);
```

Nous voyons donc ici que nous avons deux alternatives de codage pour la fonction de saisie dont l'une utilise le passage par référence. Nous pouvons également tirer profit du mécanisme du passage par référence pour gérer des situations où une fonction «retourne» plusieurs résultats. Imaginons par exemple que l'on souhaite écrire une fonction qui convertit des coordonnées cartésiennes en coordonnées polaires. La fonction prendrait donc en paramètres les deux coordonnées cartésiennes, et il s'agit à partir de ces deux coordonnées cartésiennes, de calculer des coordonnées polaires. Comme en C++, une fonction ne peut retourner qu'un seul résultat, nous pouvons prendre le parti ici de ne rien faire retourner à la fonction. Par contre, de lui fournir en paramètres deux variables passées par références qui stockeront le résultat final. Ce mécanisme est également utilisé

notes

résumé

2m 1s



par exemple pour pouvoir échanger le contenu de deux variables, le passage par référence est ici indispensable pour réaliser ce traitement. pour réaliser ce traitement.

notes

résumé

2m 49s