

Exercices

Ces exercices vous donneront des exemples d'algorithmes répondant à une problématique donnée. L'ensemble de ces algorithmes peut former une base de révision algorithmique car ce document couvre les éléments nécessaires pour construire la plupart des algorithmes procéduraux existant.

Problèmes

Vous trouverez la liste des problèmes et une de leurs solutions algorithmiques ci-dessous.

1. Echanger le contenu de 2 variables

Problème : on souhaite échanger le contenu de `U` et de `V`.

```
U <- 25
V <- 2
```

Algorithme :

```
C <- U
U <- V
V <- C
```

2. Echanger le contenu de 3 variables

Problème : on souhaite échanger le contenu de `U`, de `V` et de `Z`. C'est-à-dire de mettre la donnée de `U` dans `V`, `V` dans `Z` et celle de `Z` dans `U`.

Algorithme :

```
C <- Z
Z <- V
V <- U
U <- C
```

3. Vérifier si un utilisateur est majeur

Problème : Vérifier si un utilisateur est majeur grâce à son âge. Mettre une variable `resultat` à `Vrai` si cela est le cas, sinon mettez la à `Faux`.

```
age <- 25
```

Algorithme :

```
resultat <- Faux
@Si age >= 18
  @DebutBloc
  resultat <- Vrai
  @FinBloc
@Sinon
  @DebutBloc
  resultat <- Faux
  @FinBloc
```

ou

```
resultat <- Faux
@Si age >= 18
  @DebutBloc
  resultat <- Vrai
  @FinBloc
```

ou

```
resultat <- age >= 18
```

4. Vérifier si un utilisateur est mineur

Problème : Vérifier si un utilisateur est mineur grâce à son âge. Mettre une variable `resultat` à `Vrai` si cela est le cas, sinon mettez la à `Faux` .

```
age <- 12
```

Algorithme :

```
resultat <- age < 18
```

5. Echanger les 2 premiers éléments d'un tableau

Problème : Echanger les 2 premiers éléments d'un tableau

```
tab <- [23, 4, 2, 543, 34, ...]
```

Algorithme :

```
C <- tab[0]
tab[0] <- tab[1]
tab[1] <- C
```

6. Compter le nombre d'éléments dans un tableau

Problème : Compter le nombre d'éléments dans un tableau. Mettre le compte dans `resultat` .

```
tab <- [23, 4, 2, 543, 34, ...]
```

Algorithme :

```
resultat <- 0
@PourChaque element @Dans tab
  @DebutBloc
  resultat <- resultat + 1
  @FinBloc
```

7. Faire la somme des éléments d'un tableau

Problème : Faire la somme des éléments d'un tableau

```
tab <- [23, 4, 2, 543, 34, ...]
```

Algorithme :

```
somme <- 0
@PourChaque element @Dans tab
  @DebutBloc
  somme <- somme + element
  @FinBloc
```

8. Trouver l'élément maximum d'un tableau

Problème : Trouver l'élément maximum d'un tableau comprenant des nombres de 0 à 1000. Mettre le nombre maximum dans `resultat` .

```
tab <- [...]
```

Algorithme :

```
resultat <- tab[0]
@PourChaque element @Dans tab
  @DebutBloc
  @Si element > resultat
    @DebutBloc
    resultat <- element
  @FinBloc
@FinBloc
```

9. Trouver l'élément minimum d'un tableau

Problème : Trouver l'élément minimum d'un tableau comprenant des nombres de 0 à 1000

```
tab <- [2, ...]
```

Algorithme :

```
resultat <- tab[0]
@PourChaque element @Dans tab
  @DebutBloc
  @Si element < resultat
    @DebutBloc
    resultat <- element
  @FinBloc
@FinBloc
```

10. Trouver le premier élément supérieur à 500

Problème : Trouver le premier élément supérieur à 500. S'il n'y en a pas, le resultat doit être 0

```
tab <- [23, 4, 2, 543, 34, ...]
```

Algorithme :

11. Copier un tableau

Problème : Copier le tableau `tab` dans un autre tableau `autre`

```
tab <- [23, 4, 2, 543, 34, ...]
```

Algorithme :

12. Copier les premiers éléments d'un tableau dont la somme fait au moins 500

Problème : Copier les premiers éléments d'un tableau dont la somme fait au moins 500

```
tab <- [23, 4, 2, 543, 34, ...]
```

Algorithme :