

# Exercice 17

---

## Exercice de création de l'algorithme de programmation d'un distributeur de metro

1. ENONCE 2
2. CORRECTION 2

## 1. Enoncé

A l'image du fonctionnement du distributeur de délivrance des tickets du métro de Toulouse, écrire l'algorithme du programme qui permet de programmer un tel distributeur.

Vous êtes libre de définir le besoin et le limiter aux fonctionnalités principales du distributeur.

## 2. Correction

Cet algorithme décrit le comportement d'un programme informatique permettant l'achat d'un billet de métro avec un distributeur automatique.

Le besoin précis est celui que vous connaissez quand vous utilisez le distributeur de Tisseo dans les gares du métro.

```
Algorithme du programma principal (ALGO 1):
-----
Debut
  Initialisation du distributeur (initialisation réseau,
    accès aux données de la base, init monailleur, ....) ;

  Si Initialisation non correcte alors
    Afficher une fenetre indiquant la panne du distributeur;
    // Le programme redémarrera suite à un off/on du distributeur qui lancera
    // automatiquement le programme
  Sinon
    fini = faux;
  Tantque non fini faire
    Afficher la fenetre d'accueil (info, pub, ) contenant
      une invitation à appuyer au centre de l'écran;
  Si selection du centre de la fenetre alors
    Afficher la fenetre de choix des produit (ALGO 1.1);
    Si non annuler alors
      Afficher la fenetre du choix du paiement avec le prix;
      Si non annuler alors
        Si le choix est par CB alors Payer en CB (ALGO 1,2);
        Sinon Si le choix est par monnaie
          Alors Payer en monnaie (ALGO 1.3);
          Finsi
        Si non annuler et paiementAccepte alors
          Distribuer les produits choisis (ALGO 1.4);
          Distribuer le reçu (1.5);
        Finsi
      Finsi
    Finsi
  Finsi
Fintantque
Finsi
Fin
```

Afficher la fenetre de choix du produit (ALGO 1.1)  
(IHM synchrone)

-----  
En entree de l'algorithmme on a les données suivantes :

- les produits possibles :
  - 1 déplacement à 1,40
  - Aller/retour à 2,50
  - 10 déplacement à 11,70
  - 12 déplacements semaine à 8,60
  - Journalier à 4,00

En sortie de l'algorithmme on a les données suivantes :

- un tableau des produits choisis
- la validation ou non de la fenetre (validé ou annulé) : valide (vrai ou faux)
- le prix total des produits : prixTotal

Un produit choisi est défini par :

- son type de produit : typeProduit
- la quantité choisi : quantitéProduit

Debut

```
Creer la fenetre (cadre);
Pourchaque produit des produits possibles faire
  Si le produit est en stocke alors
    Afficher le nom du produit, son prix unitaire;
    Afficher le bouton de sélection de la quantité (0 par default);
    Associer l'action Quantité au bouton;
  Sinon
    Afficher le nom du produit en précisant non en stocke
  Finsi
Finpour
Afficher le message d'info à vide;
Afficher les boutons Valider et Annuler;
Associer l'action Valider au bouton Valider;
Associer l'action Annuler au bouton Annuler;
Afficher la fenetre;
Attendre la fermeture de la fenetre;
```

Fin

Action Quantité : cette action consiste à incrémenter un compteur de 0 à max

Debut

```
Effacer le message d'info;
max = nbre du produit en stocke;
Si sélection + alors
  Si la quantité < max alors incrémenter la quantité
Finsi
Si sélection - alors
  Si la quantité >=0 alors décrémenter la quantité
Finsi
```

Fin

```
Action Valider :
Debut
  Si aucun produit choisi (quantités à 0) alors
    Afficher le message d'info "Veuillez sélectionner un produit"
  Sinon
    prixTotal = 0;
    Pourchaque quantités sélectionnées faire
      prixTotal = prixTotal + quantité X prix unitaire du produit possible;
    Finpour
    valide = vrai;
    fermer la fenêtre;
  Finsi
Fin

Action Annuler :
Debut
  valide = faux;
  fermer la fenêtre;
Fermer

Payer en CB (ALGO 1.2)
(IHM synchrone)
-----
En entrée on a le prix total : prixTotal
En sortie on a :
  - la validation ou non de la fenetre (validé ou annulé) : valide (vrai
                                                                ou faux)
  - si le paiement a été accepté : paiementAccepte
Debut
  Afficher la fenetre d'insertion de la CB;
  Attendre la fermeture de la fenetre;
  Si non annuler alors
    Afficher la fenetre de saisie du numero de CB;
    Attendre la fermeture de la fenetre;
    Si non annuler alors
      Demander l'autorisation de paiement;
      Si Autorise alors
        Afficher la fenetre qui précise que l'autorisation a été acceptée
          et qu'il faut ne pas oublier de récupérer la CB;
        // Cette fenetre s'affiche pendant 10 secondes
        paiementAccepte = vrai;
      Sinon
        Afficher la fenetre qui précise que l'autorisation n'a pas été
          acceptée et qu'il faut ne pas oublier de récupérer la CB;
        // Cette fenetre s'affiche pendant 10 secondes
        paiementAccepte = faux;
      Finsi
    Finsi
  Finsi
  Finsi
  Finsi
Fin
```

Payer en monnaie (ALGO 1.3)  
(IHM synchrone)

-----  
En entrée on a le prix total : prixTotal  
En sortie on a :  
- la validation ou non de la fenetre (validé ou annulé) : valide (vrai ou faux)  
- si le paiement a été accepté : paiementAccepte  
On affiche une fenêtte qui contient la somme à payer et la somme restante à payer qui est mise à jour au fur et à mesure que l'on insert des pièces dans le monailleur.  
Elle contient un bouton Annuler (pas de bouton validation car la validation est automatique quand la somme insérée est égale ou dépasse la somme à payer).

Debut

```
Creer la fenetre (cadre);  
Afficher la somme à payer : sap;  
Afficher la somme restante à payer : srap;  
Afficher le bouton Annuler;  
Associer l'action AnnulerMonnaie au bouton Annuler;  
Initialiser le monailleur;  
Afficher l'action PieceInserée au monailleur;  
Afficher la fenêtre;  
Attendre la fermeture de la fenêtre; // La fenetre se ferme si srap est <= 0  
Si annulation alors  
    rendre les pieces insérées  
Sinon  
    Si srap < 0 alors  
        rendre srap en pieces;  
    Finsi  
    paiementAccepte = vrai;  
Finsi
```

Fin

Action PieceInserée :

Debut

```
Si la piece est non reconnue alors  
    rendre la piece  
Sinon  
    Soit val la valeur de la pièce détectée par le monailleur;  
    srap = srap - val;  
    Mise à jour de l'affichage de la somme restante à payer avec srap;  
    Si srap <=0 alors  
        Fermer la fenetre  
    Finsi
```

Finsi

Fin

Distribuer les produits choisis (ALGO 1.4)

-----  
Debut

```
Pour chaque produit choisi faire  
    Marquer sur le billet la date courante;  
    Sortir le billet;  
Finpour
```

Fin

```
Distribuer le reçu (ALGO 1.5)
```

```
-----  
Debut
```

```
  Si paiement par CB alors
```

```
    Editer le reçu avec les informations de la CB, la date et le prixTotal;
```

```
  Sinon
```

```
    Editer le reçu avec les informations : date, prixTotal et somme rendus;
```

```
  Finsi
```

```
Fin
```