TP4-IFT3913-Automne 2023 Auteurs:

Kanty-Louange Gakima (20184109)

Yann-Sibril Saah (20061840)

**Rapport**

**1. TESTS BOITE NOIRE**

On suppose que la **spécification** du « Currency Converter » exige que:

* + Il doit convertir des montants entre les devises suivantes : USD, CAD, GBP, EUR, CHF, AUD.
  + Il doit seulement accepter des montants entre [0, 1 000 000].

On aura donc 5 classes d’équivalences :

D1 = {USD, CAD, GBP, EUR, CHF, AUD} pour les devises valides

D2 = {USD, CAD, MP, EUR, CHF, RP, CNY} qui contient des types de devises invalides

MP: Mexican Pesos, RP: Roupie, CNY: Chinese Yuan Renminbi

D3 = {0<= d <= 100 000} pour les valeurs appartenant à l’intervalle du montant accepté

D4 = {d < 0} pour les valeurs correspondant à un montant inférieur à celui accepté

D5 = {d > 100 000} pour les valeurs correspondant à un montant supérieur à celui accepté

**Jeu de test :**

**Tests de la méthode currencyConverter.MainWindow.convert :**

- Test pour les devises valides:

("Euro", "US Dollar", currencies, 4000) Réussi

- Test pour une devise de départ invalide :

("Mexican Pesos", "US Dollar", currencies, amount) Échoué

- Test pour une devise d’arrivée invalide :

("US Dollar", "Chinese Yuan Renminbi", currencies, amount) Échoué

-Test de la conversion de 2 devises invalides :

("Roupie", "Chinese Yuan Renminbi", currencies, amount) Échoué

**Tests de la méthode currencyConverter.Currency.convert :**

- Test pour les valeurs appartenant à l’intervalle du montant accepté

(4000, 1.35) : Réussi

- Test d’un montant valide à la borne inférieure :

(0, 1.335) : Réussi

- Test d’un montant invalide proche de la borne inférieure :

(-1.0, 1.335) : Échoué

- Test d’un montant invalide éloignée de la borne inférieure :

(-17600, 1.335) : Échoué

- Test d’un montant valide à la borne supérieure :

(1000000, 1.335) : Réussi

- Test d’un montant invalide proche de la borne supérieure :

(1000001, 1.335) : Échoué

- Test d’un montant invalide éloignée de la borne supérieure:

(2000000, 1.335) : Échoué

**Observation des résultats :** On a un total de **11 tests**, **7 échecs** et **4 succès.**

**Hypothèse pour une entrée invalide** : Selon la spécification on a pris comme hypothèse que si une devise ou un montant ne respecte pas la spécification le code ne renvoie aucun résultat mais plutôt une exception lors de tests.

**Observation** : On remarque que pour la méthode currencyConvert.MainWindow.convert , la spécification n’est pas respectée et les tests échoue pour toutes les entrées invalides, Pour éviter cela il faudrait vérifier si la saisie est valide selon la spécification et lancer une exception sinon

**2. TESTS BOITE BLANCHE :**

**Jeu de test :**

**Critère de couverture d’instructions :**

* currencyConvert.MainWindow.convert :

ShortNameCurrency2 == Null : currency2 invalide:

("Euro", "Mexican Pesos », currencies, amount) : Réussi

ShortNameCurrency != Null: les 2 devises sont valides

("Euro", "US Dollar", currencies, amount) : Réussi

* currencyConvert.Currency.convert :

(50, 1.50) Réussi

**Critère de couverture des arcs du graphe de flot de contrôle :**

* **Méthode currencyConvert.MainWindow.convert** :

**Arc 1 : les 2 devises sont valides et se trouvent dans la liste :**

("Euro", "US Dollar", currencies, 50): Réussi

**Arc2: la devise d’arrivée est invalide:**

("Euro", "", currencies, amount): Réussi

**Arc3 : la devise de départ est invalide :**

("Roupie", "US Dollar", currencies, amount): Réussi

* Pour la méthode currencyConvert.Currency.convert  on a un seul arc et il a été déjà testé au critère de couverture d’instructions

**Critère de couverture des chemins indépendants du graphe de contrôle :** complexité cyclomatique = 5

**Chemin1 : les 2 devises sont valides et se trouvent dans la liste :**

("Euro", "US Dollar", currencies, 50): Réussi

**Chemin2 : la devise d’arrivée est invalide:**

("Euro", "Mexican Pesos », currencies, amount) : Réussi

**Chemin3 : la devise de départ est invalide :**

(“Roupie”, “US Dollar”, currencies, amount): Réussi

**Chemin4 : les deux devises sont invalides**

**(**"Roupie", "Mexican Pesos", currencies, 4000.0) : Réussi

Impossible d’avoir un 5 ème chemin car tous les chemins ont été couverts

**Critère de Couverture de Conditions :** Les deux méthodes n’ont pas de conditions composées donc pas de tests pour ce critère

**Couverture des i-chemins :**

- Méthode currencyConvert.MainWindow.convert :

Selon la structure du code la couverture des i-chemins revient à faire les mêmes tests que pour le critère de couverture des chemins indépendants du graphe de contrôle

Et pour la méthode currencyConvert.Currency.convert  on a un seul chemin qui a été montré au critère de couverture d’instructions

**Observation des résultats :** tous les tests structurels passent avec succès.