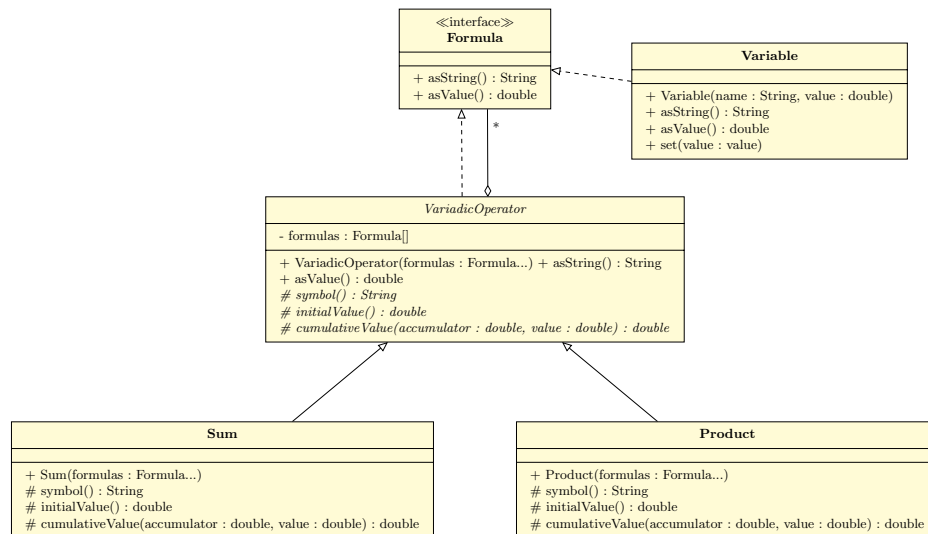


POO – TD 8

Affichages des formules :

Nous souhaitons pouvoir traduire les formules sous des formats différents (XML, Texte, \LaTeX , etc.). Nous supposons que les classes et l'interface présentes sur le diagramme suivant sont déjà implémentées. Vous remarquerez qu'aucune méthode d'affichage n'est proposée sur ce diagramme.



1. Proposez une implémentation des différents affichages qui ne nous demandent pas de créer de nouvelles classes. Dans quelles classes se situe le code des méthodes d'affichage ? Est-ce satisfaisant ?
2. Les diagrammes 1 et 2 proposent deux façons de résoudre le problème de conception identifié à la question précédent. Les méthodes d'affichage sont regroupées dans des classes séparées appelées **Visitor**. Chaque visiteur va "visiter" les classes qui implémentent l'interface **Formula** afin de générer une chaîne de caractères les décrivant. Quelles sont les différences entre

les versions cyclique et acyclique du patron de conception **Visitor** ? Notez que, dans les deux versions, l'interface **Formula** possède une méthode **accept(FormulaVisitor<R> visitor)** : une implémentation de cette méthode dans une classe **C** doit appeler la méthode **visit(C c)**.

3. Implémentez la version acyclique du patron de conception **Visitor**. La classe **PlainTextVisitor** doit construire la chaîne $((b+c)*(a+b))$ pour une formule **f** obtenue de la façon suivante :

```

Variable a = new Variable("a", 0);
Variable b = new Variable("b", 0);
Variable c = new Variable("c", 0);
Formule formula = new Product(new Sum(a,b), a);
    
```

La classe **XMLVisitor** doit produire le document XML suivant (sans l'indentation et les espaces pour simplifier l'implémentation de la classe) :

```

<product>
  <sum>
    <var>a</var>
    <var>b</var>
  </sum>
  <var>a</var>
</product>
    
```

4. Ajoutez une classe **SquareRoot** au module gérant les formules : cette classe calcule la racine carrée d'une autre formule. Que doit-on modifier pour que ce nouvel opérateur soit traité correctement par tous les visiteurs ?
5. Ajoutez un nouveau visiteur **LaTeXVisitor** qui traduit les formules en \LaTeX . Dans un document \LaTeX , la formule $\sqrt{((a \times b) + (c \times d))}$ s'écrit \LaTeX `\sqrt{((a \times b) + (c \times d))}`.
6. Ajoutez un nouveau visiteur **EvaluatorVisitor** qui permet d'évaluer les formules afin de remplacer les implémentations des méthodes **asValue()** par un visiteur. Les méthodes **visit** doivent retourner les résultats des évaluations.

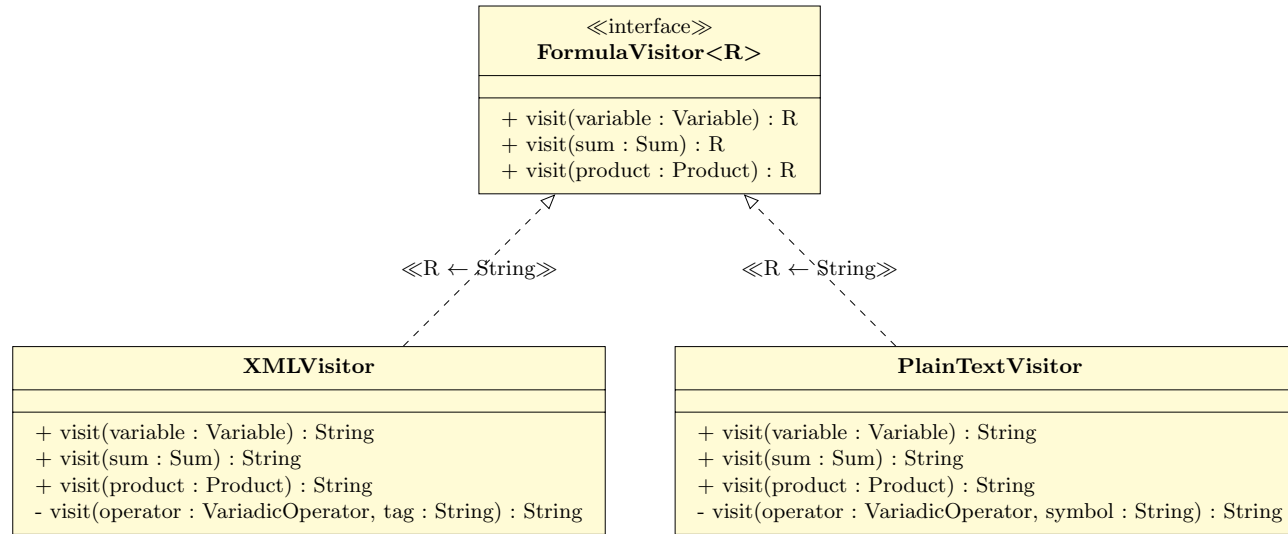


Diagramme 1 – Version cyclique

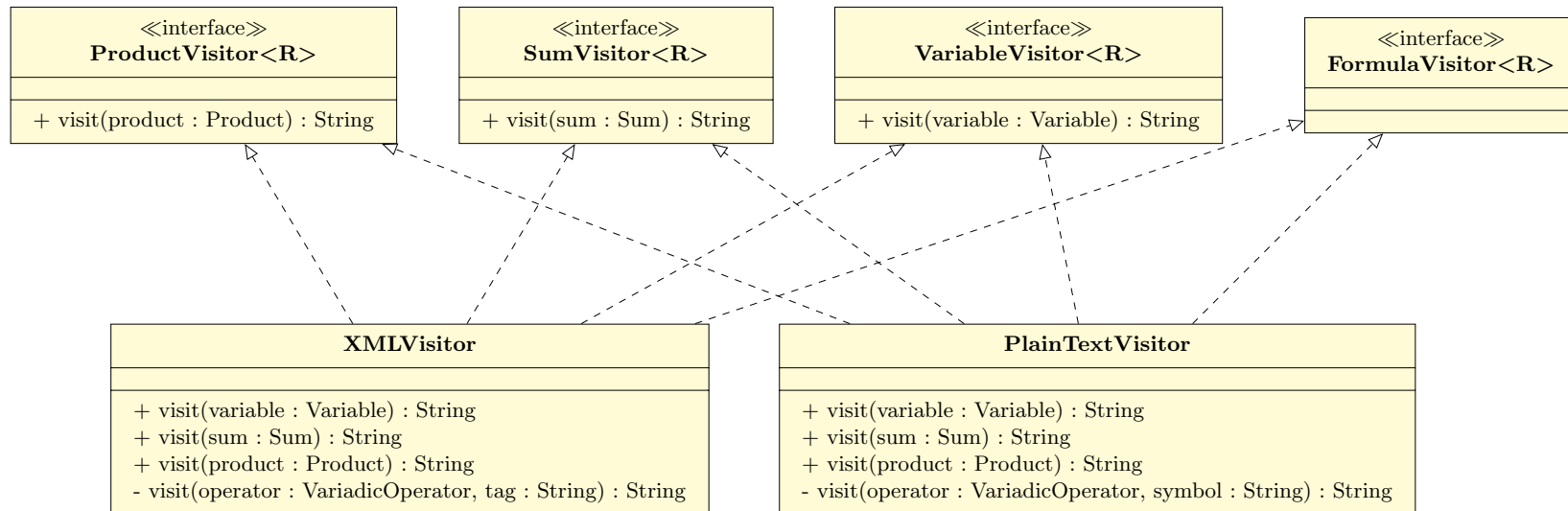


Diagramme 2 – Version acyclique