

NOM et Prénom : _____
Groupe : L2 - A

1 Rust syntaxe, typage, concepts de base

1. Ce code peut-il compiler ? Pourquoi dans les deux cas :

```
let mut a = vec![1, 2, 4, 5, 12];
let mystère = &a[4];
a.pop();
println!("{}", mystère);
```

- `&a[i]` correspond à `fn index(Vec<T>, index: usize) -> &T`
- `pub fn pop(&mut self) -> Option<T>`
- `T` est un `i32`

2. Quel type représente une valeur qui peut-être absente en Rust. Donner une définition de ce type en Rust et un exemple d'usage.

3. Soit le code suivant :

```
let a = vec![1, 42, 4, 1];  
let b = a;  
println!("{}", a);
```

Est-ce que ce code compile ? Pourquoi dans les deux cas ?

4. À quoi sert le mot clef `match` ? À quoi sert une `enum` ? Donner le cas obligatoirement pour une `enum` et éventuellement les autres cas.
5. Expliquer ce qu'est l'immuabilité, les références et les règles importantes de partage associées et le concept d'ownership (propriété). Bonus : Tentez d'expliquer quels bugs sont exclus grâce à ces règles et dans quel cas sur des processeurs modernes cela peut-être pratique.