

Lifting State Up

Parfois des composants doivent faire remonter des données dynamiques à leur parent direct. Nous allons découvrir cette technique dans un exercice d'application.

1. Exercice Partie 1/2

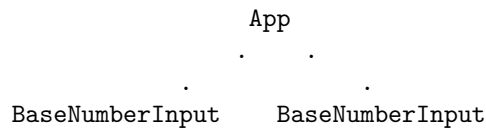
Vous allez créer un script React permettant de réaliser une conversion de nombre décimal vers un nombre binaire.

Créez deux composants **App** et **BaseNumberInput**. Le composant **App** contiendra deux composants **BaseNumberInput** : respectivement pour saisir le nombre décimal et afficher le nombre binaire.

Nombre décimal : [...]

Nombre binaire : [...]

Hiérarchie des composants :



Dans le composant **BaseNumberInput** l'attribut `onChangeBase` sera appelé dans le composant lui-même. La logique algorithmique du changement de base sera implémentée dans le composant parent **App** (méthode `this.handleChange`) :

```
<BaseNumberInput onChangeBase={(e) => this.handleChange(e)} />
```

```
// Dans le composant BaseNumberInput
this.props.onNumberChanger(e.target.value)
```

1. Exercice Partie 2/2

Implémentez la logique dans l'autre sens : si on rentre un nombre binaire dans le champ correspondant alors la conversion se fait dans l'autre sens (binaire vers décimal).

Number: decimal

12

Number: binary

1100

Figure 1: Conversion binary \leftrightarrow decimal