

TD 2 : Codage dense

Vous allez utiliser le simulateur VQS pour réaliser une simulation de **codage dense**.

Très important : **commencez par mettre à jour votre distribution de VQS via GIT.**

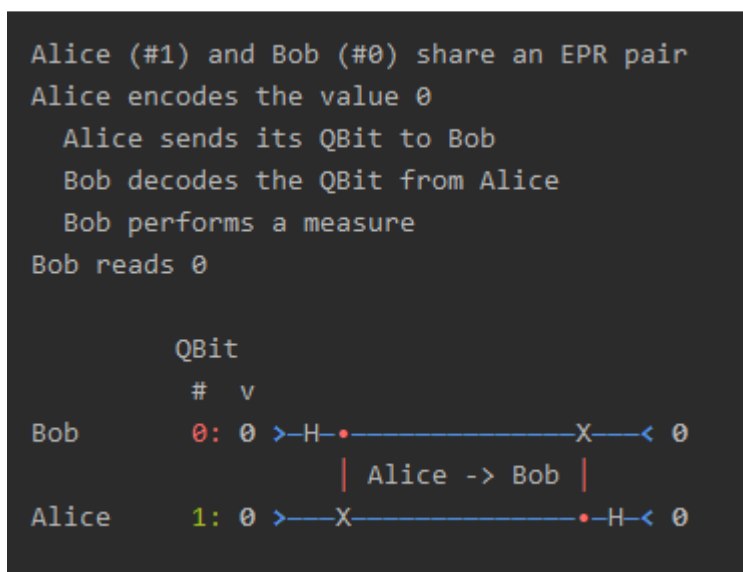
(si vous ne connaissez pas suffisamment GIT, placez vous dans un nouveau répertoire et rechargez l'espace de projet par **git clone <https://github.com/gmenier/VisualQuantumSimulator.git>**, puis repartez du main comme pour le premier TD)

Vous allez réaliser une fonction

```
def superDense(vcode : Int) {
}
```

... qui permet de simuler la transmission de la valeur vcode.

Par exemple, **superDense(0)** donne l’affichage suivant :



Notez que vous pouvez également utiliser $C(X(0), 1)$ (et d'autres paramètres)

superDense(1) donne dans la console :

```
Alice (#1) and Bob (#0) share an EPR pair
Alice encodes the value 1
  Alice sends its QBit to Bob
  Bob decodes the QBit from Alice
  Bob performs a measure
Bob reads 1

      QBit
      #  v
Bob    0: 0 >-H-•-----X-< 1
          |      Alice -> Bob |
Alice  1: 0 >---X-X-----•-H-< 0
```

superDense(2) :

```
Alice (#1) and Bob (#0) share an EPR pair
Alice encodes the value 2
  Alice sends its QBit to Bob
  Bob decodes the QBit from Alice
  Bob performs a measure
Bob reads 2

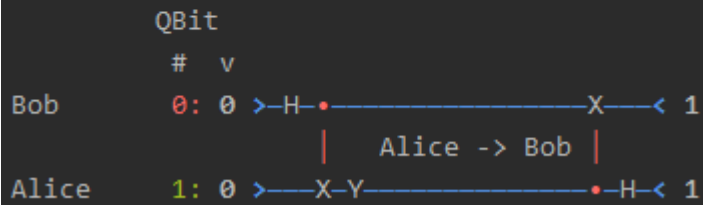
      QBit
      #  v
Bob    0: 0 >-H-•-----X-< 0
          |      Alice -> Bob |
Alice  1: 0 >---X-Z-----•-H-< 1
```

superDense(3) :

```

Alice (#1) and Bob (#0) share an EPR pair
Alice encodes the value 3
  Alice sends its QBit to Bob
  Bob decodes the QBit from Alice
  Bob performs a measure
Bob reads 3

```



Pour effectuer une action en fonction d'une valeur, vous pouvez utiliser `if () { } else { }` (même syntaxe que Java).

Rappel : pour provoquer une mesure, `rr - <(0)` ou `rr - <(1)` puis pour demander la valeur par `rr.?(0)` ou `rr.?(1)`. Notez que vous pouvez provoquer la lecture de tous les QBits du registre en utilisant `rr - <()` puis `rr. ?()` qui vous donne la valeur décimale de la lecture (donc ici, 0, 1, 2 ou 3).

Utilisez la visualisation des images en cercle pour bien décoder les étapes des calculs. C'est très important de bien comprendre la séquence de transformation des états. En particulier, ne confondez pas les états (parmi 4 états possibles) du couple de QBit (c'est ce qui est affiché par les cercles) et les états séparés des deux QBits (ce n'est pas affiché par les images cercles).

Vérifiez que vous êtes capable de calculer à la main (avec des produits tensoriels et les opérateurs quantiques) la séquence des états et vérifiez le codage en cercle.

Le circuit et les CNot ne semble pas être dans le même sens que dans le cours : est-ce gênant ? Refaites les calculs du cours et vérifiez que vous obtenez bien les mêmes résultats. Qu'est-ce qui change ?