

Le jeu du plus ou moins

But du jeu : le joueur doit deviner un nombre compris entre 1 et 100.

1. Création des variables :

L'élève devra réfléchir aux nombres de valeurs qui interviennent.

Il faut deux variables minimum :

- NbCherche : correspondant au nombre à chercher
- NbDonne : correspondant au nombre donné par le joueur

2. Introduction du nombre à chercher

On se fixe dans un premier temps le nombre à chercher.

Exemple : NbCherche prend la valeur 10.

3. Nombre donné par l'utilisateur

Il faut vérifier que le joueur rentre bien une valeur entre 1 et 100.

Cela permet de retravailler la notion de négation (la négation de *et* est *ou*)

Proposition d'algorithme :

```
Début
    NbCherche prend la valeur 10
    Afficher "Entrer un nombre compris entre 1 et 100"
    Saisir NbDonne
    Tant que (NbDonne < 1 ou NbDonne > 100) faire
        Afficher "Entrer un nombre compris entre 1 et 100"
        Saisir NbDonne
    FinTantque
fin
```

4. Première condition

On demande à l'élève d'écrire les conditions pour afficher "plus" ou "moins".

On met en évidence qu'il faut faire attention au niveau du *sinon*, pour exclure l'égalité.

Proposition d'algorithme :

```
Début
    NbCherche prend la valeur 10
    Afficher "Entrer un nombre compris entre 1 et 100"
    Saisir NbDonne
    Tant que (NbDonne < 1 ou NbDonne > 100) faire
        Afficher "Entrer un nombre compris entre 1 et 100"
        Saisir NbDonne
    FinTantque
    Si (NbDonne < NbCherche)
        Alors
            Afficher "C'est plus"
        Sinon
            Afficher "C'est moins"
    FinSi
fin
```

5. Répéter tant qu'on n'a pas le bon nombre

Il sera intéressant de voir à quel endroit, il sera judicieux de placer la boucle *Tant que*.

Proposition d'algorithme :

```
Début
    NbCherche prend la valeur 10
    Afficher "Entrer un nombre compris entre 1 et 100"
    Saisir NbDonne
    Tant que (NbDonne ≠ NbCherche) faire
        Tant que (NbDonne < 1 ou NbDonne > 100) faire
            Afficher "Entrer un nombre compris entre 1 et 100"
            Saisir NbDonne
        FinTantque
        Si (NbDonne < NbCherche)
            Alors
                Afficher "C'est plus"
            Sinon
                Afficher "C'est moins"
        FinSi
        Afficher "Entrer un nombre compris entre 1 et 100"
        Saisir NbDonne
    FinTantque
    Afficher "Vous avez trouvé !"
fin
```

6. Finalisation

On donne une valeur aléatoire à la variable NbCherche.

```
NbCherche prend la valeur aléatoire compris entre 1 et 100
```

7. Compteur (facultatif)

Ajouter un compteur, comptant le nombre de test, même erroné.

Proposition d'algorithme :

```
Début
    NbCherche prend la valeur aléatoire compris entre 1 et 100
    Afficher "Entrer un nombre compris entre 1 et 100"
    Saisir NbDonne
    Compteur prend la valeur 1
    Tant que (NbDonne ≠ NbCherche) faire
        Tant que (NbDonne < 1 ou NbDonne > 100) faire
            Afficher "Entrer un nombre compris entre 1 et 100"
            Saisir NbDonne
            Compteur prend la valeur Compteur +1
        FinTantque
        Si (NbDonne < NbCherche)
            Alors
                Afficher "C'est plus"
            Sinon
                Afficher "C'est moins"
        FinSi
        Afficher "Entrer un nombre compris entre 1 et 100"
        Saisir NbDonne
        Compteur prend la valeur Compteur +1
    FinTantque
    Afficher "Vous avez trouvé le nombre en : "
    Afficher Compteur
    Afficher "coups"
fin
```

8. Limiteur de coup (facultatif)

L'intérêt de créer une variable NbCoup indiquant le nombre de coups permet dans le futur de modifier juste cette variable sans avoir à relire tout l'algorithme. Cela se fait régulièrement en programmation.

On peut également indiquer au joueur le nombre de tests qui lui reste.

Proposition d'algorithme où l'on compte les nombres non compris dans [1 ; 100] :

```
Début
  NbCoup prend la valeur 10
  NbCherche prend la valeur aléatoire compris entre 1 et 100
  Afficher "Entrer un nombre compris entre 1 et 100"
  Saisir NbDonne
  Compteur prend la valeur 1
  Tant que ((Compteur ≤ NbCoup) ou (NbDonne ≠ NbCherche)) faire
    Tant que (NbDonne < 1 ou NbDonne > 100) faire
      Afficher "Il vous reste :"
      Afficher NbCoup-Compteur
      Afficher "possibilités pour trouver le nombre"
      Afficher "Entrer un nombre compris entre 1 et 100"
      Saisir NbDonne
      Compteur prend la valeur Compteur +1
    FinTantque
    Si (NbDonne < NbCherche)
      Alors
        Afficher "C'est plus"
      Sinon
        Si (NbDonne > NbCherche)
          Alors
            Afficher "C'est moins"
          FinSi
        FinSi
      Afficher "Il vous reste :"
      Afficher NbCoup-Compteur
      Afficher "possibilités pour trouver le nombre"
      Afficher "Entrer un nombre compris entre 1 et 100"
      Saisir NbDonne
      Compteur prend la valeur Compteur +1
    FinTantque
    Si (Compteur ≤ NbCoup)
      Alors
        Afficher "Vous avez trouvé le nombre en :"
        Afficher Compteur
        Afficher "coups"
      Sinon
        Afficher "Vous avez perdu.
          Vous n'avez plus de possibilité pour trouver le nombre"
    fin
```