

B1_BDP_M2_20231123

Partie I

On rappelle que la **sortie** d'un programme consiste en ce qui va s'afficher à l'écran lors de son exécution *et rien d'autre*.

Donnez la sortie du programme suivant:

JAVA

```
int nbEntier;
nbEntier = 1;
String s = "hello";
int nb1 = 5;
if (nbEntier > 2) {
    System.out.print("ici");
    nbEntier = nbEntier + 2;
    System.out.println(nbEntier);
}
if (nbEntier <= 2) {
    System.out.print("là");
    nbEntier = nbEntier + 5;
    System.out.println(nbEntier);
} else {
    System.out.println("encore");
    nbEntier--;
    System.out.println(nbEntier);
}
System.out.println(nbEntier);
int nb2 = 0;
for (int i = 1; i <= nb1; i++) {
    System.out.println(s + i + " : " + nb2);
    nb2 = nb2 + 10;
}
System.out.println(nb2);
```

Partie II

On considère la version 3 du *Jeu de l'oie* vue en cours et dont une implémentation correcte est reproduite ci-dessous:

JAVA

```
01 int caseObjectif = 20;
02 Random generateur = new Random();
03 int caseCourante = 0;
04 int compteurLancers = 0;
05 boolean gagne = false;
06
07 while (!gagne) {
08     int lancer = generateur.nextInt(6) + 1;
09     compteurLancers++;
10     caseCourante = caseCourante + lancer;
11     if (caseCourante > caseObjectif) {
12         int depassement = caseCourante - caseObjectif;
13         caseCourante = caseObjectif - depassement;
14     }
15     System.out.println(String.format("Lancer %d : vous avez fait %d. Vous êtes sur la case %d.",
16         compteurLancers, lancer, caseCourante));
17     if (caseCourante == caseObjectif) {
18         System.out.println("Vous avez gagné !");
19         gagne = true;
20     } else {
21         System.out.println("Il vous reste " + (caseObjectif - caseCourante) + " cases.");
22     }
23 }
```

01 Expliquez la **différence entre les opérateurs = et ==** . Vous pouvez vous appuyer sur le code pour illustrer votre propos.

02 Expliquez le rôle de la variable `gagne` **en vous appuyant sur toutes ses apparitions** dans le programme.

On vous demande d'implémenter de **nouveaux besoins** pour ce programme. Vous considérerez que chaque question s'appuie sur la version de base ci-dessus (indépendamment des autres questions). Pour chaque implémentation d'un besoin, vous ne réécrirez pas l'ensemble du programme : vous n'écrirez que ce qui est modifié/ajouté, en indiquant à chaque fois les numéros de lignes de code concernées (par exemple: «*remplacer la ligne 17 par: ...* »). Les besoins demandés sont exprimés dans les questions qui suivent.

03 L'utilisateur indique en début de programme sur quelle case il veut démarrer la partie.

04 On lance à chaque fois **deux dés** au lieu d'un seul pour savoir de combien on avance.

05 On veut maintenant pouvoir jouer à **deux joueurs**. Le premier joueur à atteindre la case objectif a gagné.

06 Un score est attribué à chaque partie. Ce score est déterminé par **le nombre total de cases parcourues** lors de la partie (y compris quand on «rebondit» sur la case 20). Le score de la partie doit être affiché à la fin.

Exceptionnellement, le besoin suivant s'appuie sur le précédent dont on supposera qu'il a été correctement implémenté. On veut maintenant effectuer un calcul statistique pour estimer le **nombre moyen de cases à parcourir pour gagner** (c'est-à-dire le score moyen). Pour cela, on va lancer 10 000 simulations et calculer la moyenne des scores. On rappelle que la moyenne sera obtenue en divisant la somme des scores par le nombre de simulations effectuées. La moyenne sera affichée à la fin du programme.

07 Faites une **analyse** très succincte de ce nouveau besoin. Tranchez sur les éventuels **choix d'implémentation** qui s'imposent.

08 Implémentez ce nouveau besoin.

Annexe - Documentation

```
// Afficher « Bonjour » à l'écran puis passer à la ligne
System.out.println("Bonjour");

// Afficher « Bonjour » à l'écran sans passer à la ligne
System.out.print("Bonjour");

// Récupérer une entrée utilisateur (de type entier) au clavier
Scanner clavier = new Scanner(System.in);
int entree = clavier.nextInt();

// Générer un entier aléatoire entre 0 et 99
Random generateur = new Random();
int auHasard = generateur.nextInt(100);

// Formater une chaîne de caractères - %d représente un entier
// Chaque %d sera remplacé par la valeur de la variable correspondante dans la liste qui suit la chaîne
String.format("J'ai %d pommes et %d poires.", 6, 3); // produira la chaîne : "J'ai 6 pommes et 3 poires."
```

JAVA