

Introduction à la Programmation (IP1)

Partiel (Sujet B) – Durée : 2 heures

Université Paris-Diderot – Samedi 25 octobre 2014

- Aucun document ni aucune machine ne sont autorisés. Les téléphones doivent être rangés.
- Les exercices sont tous indépendants.
- Une réponse peut utiliser les réponses attendues à une question précédente (même si elle est non traitée).
- Les fragments de code Java doivent être correctement indentés.

On rappelle que le type `boolean` est constitué des deux valeurs `true` et `false`.

Exercice 1

1. Remplir les pointillés avec un type cohérent.

(Sachant « `public static void printString (String a);` » et « `public static void printInt (int a);` ».)

```
1 public static ... f (... s, ... t) {
2     printString (s + "a");
3     printInt (t);
4 }
```

2. Donner 5 erreurs dans la fonction suivante en les expliquant à l'aide d'une courte phrase.

```
1 public static int doubleIfOdd (int n {
2     if (n % 2 = 0) {
3         return n;
4     } else if {
5         (n * "2");
6     }
7 }
```

□

Exercice 2

1. Que valent les variables `s` et `u` à la fin de l'exécution du fragment de code suivant ?

```
1 int s = 0;
2 String u = "";
3 for (int i = 0; i < 4; i++) {
4     if (i == 3) {
5         u = u + " ! Je suis ";
6     } else {
7         if (i == 4) {
8             u = u + "mé";
9         } else {
10            u = u + "ah";
11        }
12        s = s + i;
13    }
14 }
15 u = u + "content.";
16 printString(" Et toi?");
```

2. Qu'affiche ce programme ?

□

Exercice 3

1. Un entier m est divisible par un entier k si le reste de la division entière de m par k est nul. On rappelle qu'un nombre premier est un entier strictement plus grand que 1 et qui est divisible uniquement par 1 et par lui-même.

Pourquoi la fonction suivante n'est-elle pas correcte ? On souhaiterait en effet qu'elle renvoie `true` si et seulement si son argument n est premier. (On suppose n positif.)

```

1 public static boolean isPrime (int n) {
2     boolean b = false;
3     for (int i = 2; i < n; i++) {
4         if (n % i == 0) {
5             b = true;
6         }
7     }
8     return b;
9 }
```

2. Que renvoie-t-elle précisément, en fonction de n ?
3. Comment modifier cette fonction de façon à ce qu'elle renvoie effectivement `true` si et seulement son argument est un nombre premier ?

□

Exercice 4

1. Écrire un programme qui affiche le motif suivant répété sur 300 caractères de large.

```

1 *   *   *
2 * * * * *
3 *   *   *
```

Fonctions et procédures pouvant être utilisées :

– « `public static void printString (String s);` » affiche la chaîne s à l'écran.

□

Exercice 5

1. Écrire une fonction qui prend en argument un tableau d'entiers et qui renvoie un tableau contenant les mêmes éléments en ordre inverse. Par exemple, sur le tableau $\{1, 2, 3, 4\}$, la fonction renverra le tableau $\{4, 3, 2, 1\}$.

Fonctions et procédures pouvant être utilisées :

– « `public static int intArrayLength(int[] t);` » retourne la longueur du tableau t .

□

Exercice 6

1. Écrire une fonction `isAlmostPerfect` qui prend en argument un entier n (supposé positif) et qui renvoie `true` si et seulement si n est presque parfait (c'est-à-dire si la somme de tous ses diviseurs vaut $2n - 1$). On rappelle qu'un nombre k est un diviseur d'un nombre m si le reste de la division de m par k est nul.
2. Écrire une fonction `pow2` qui prend en argument un entier k , supposé positif, et qui renvoie l'entier 2^k .
3. En se servant des fonctions précédentes, écrire une fonction `pow2IsAlmostPerfect` qui prend en argument un entier k , supposé positif, et qui renvoie `true` si et seulement si l'entier 2^k est presque parfait.

□