Partiel 2 PFON ING1 2017 (mars 2015)
Qu'une seule réponse possible par question

1. Une approche fonctionnelle est plutôt orientée :
   a. bottom-up
   b. left-right
   c. top-down
   d. sud-ouest

2. Qu'appelle-t-on "lambda expression" ?
   a. une expression ordinaire
   b. une expression dénotant une fonction anonyme
   c. une expression du second ordre
   d. une fonction nommée "lambda"

3. En programmation fonctionnelle pure :
   a. il n'y a que des constantes
   b. il n'y a que des fonctions constantes
   c. il n'y a pas de parallélisme
   d. tout programme qui compile est correct

4. L'une des fonctionnalités ci-dessous n'est pas idiomatiquement non-stricte. Laquelle ?
   a. les entrées-sorties
   b. les opérateurs logiques
   c. les branchements conditionnels
   d. la gestion des exceptions (catch/throw)

5. En Haskell, il est possible d'utiliser la notation préfixe pour les opérateurs infixes. L'expression "a + b" s'écrit :
   a. (+) a b
   b. [+] a b
   c. '+' a b
   d. +(a b)

6. En Lisp, quel est la valeur l'expression suivante ? :
(labels ((foo (x) (* 2 x))) (foo "bar"))
    a. "barbar"
    b. un résultat indéterminé
    c. aucune valeur, mais une erreur de typage
    d. ne compile même pas

7. En Haskell, quelle est la valeur de ssq 1 pour ssq définit comme suit ?
ssq : : Int -> Int
ssq n = n * ssq (n - 1)
ssq 1 = 1
    a. 1
    b. stack overflow
    c. syntax error
    d. 0

8. En Haskell, que vaut l'expression double 3 + 4.0 pour double définie comme suit ?
double : : Int -> Int
double x = 2 * x
    a. 10
    b. 10.0
    c. erreur de type
    d. 14

9. En Haskell, que représente [t] ?
    a. une liste composée de la valeur booléen vrai
    b. un tableau de dimension t
    c. une liste de t éléments
    d. une liste d'éléments de type t, pour tout t

10. En Haskell, que vaut l'expression [x == 3 | x <- [2, 3, 4]] ?
    a. [False, True, False]
    b. [3]
    c. [True]
    d. 3

11. En Haskell, les chaînes de caractères sont implémentées comme :
    a. des listes de caractère
    b. des tableaux de caractère
    c. des vecteurs de caractère
    d. des types natifs

12. Quelle est la valeur de l'expression (car (list 1 (error "Barf!"))) en Lisp ?
    a. 1
    b. SIMPLE-ERROR : Barf!
    c. 1SIMPLE-ERROR : Barf!
    d. erreur de typage : liste mal construite

13. En Haskell, que vaut head [1, error "Barf!"]
    a. 1
    b. Program error : Barf!
    c. 1Program error : Barf!
    d. erreur de type

14. En Lisp, que vaut (quote ABC) ?
    a. "ABC"
    b. la liste de 3 char A, B et C
    c. la valeur du symbole ABC, ou erreur s'il n'existe pas
    d. le symbole ABC

15. En Haskell, que vaut let x = 5 in foo 1 sachant :
x = 4
foo a = x
    a. 1
    b. 4
    c. Undefined var a
    d. 5

16. Application partielle en Haskell ?
17. En Haskell, que représente l'exp \n -> 2 * n ?
18. "Réduction" dans une langage fonctionnel ?
19. Que signifie en Haskell a . b ?
20. En Lisp, que vaut ((lambda (x) (lambda (n) (+ n x))) 5) ?