

Nom :

Prénom :



**Informatique et systèmes
de communication ISC**



Haute Ecole d'Ingénierie
Hochschule für Ingenieurwissenschaften

TEST INTERMÉDIAIRE SOLUTION

101.1 – Programmation impérative

Consigne :

Lisez attentivement la donnée et répondez de manière **lisible** aux questions. Vous avez droit pour cet examen à un aide-mémoire de 1 pages (1 feuille recto). Aucun moyen électronique n'est permis.

Un conseil : ne restez pas bloqués sur une question. Répondez tout d'abord aux questions avec lesquelles vous êtes à l'aise et revenez ensuite aux questions posant problème. Le barème indiqué est indicatif.

Question	Points	Bonus	Score
Short questions	10	2	
Loops analysis	8	0	
EBNF grammars	6	0	
Code comprehension	3	0	
Writing functions	7	1	
String manipulations	11	0	
Too fast, too furious	5	0	
Total:	50	3	

This exam has 7 questions, for a total of 50 points.

Rev 1.04ω

Question 1 – Short questions (10 points)

Cette question est séparée en plusieurs exercices indépendants. Le nombre de point pour chaque exercice est indiqué dans la marge.

[4 Pt] (a) Soient les déclarations suivantes :

```
1 val foo : Int = 0xBABA
2 val foobar : Float = 3.825f
3 val baz : String = "y"
```

Les expressions suivantes sont correctes. Donnez le **type** et la **valeur** des expressions suivantes.

1) (foobar + - foobar).toByte

1) Byte, 0

2) (foo & 0xFF0).toHexString

2) String, "ab0"

3) (((foo >> 4) << 4) | 0xE).toHexString

3) String, "babe"

4) (foo^foo).toShort

4) Short, 0

5) ((foobar * 100).toInt / 100.0f)

5) Float, 3.82f

6) if(foo < 0xFFFF) 'a' else 'b'

6) Char, 'a'

7) if(true) '1' + baz else baz + '1'

7) String, "1y"

8) baz + ('d' + 1.9).toChar + ('t'-1).toChar

8) String, "yes"

[1 Pt] (b) Quel est le contenu de r après l'exécution du code ci-dessous :

```
1 val s: String = "TikTok"
2 val r: String = StringUtils.charAt(s, StringUtils.length(s)-2) +
3 "" + StringUtils.charAt(s, 2)
```

Solution: "ok"

[1 Pt] (c) Soit le code suivant :

```
1 val a : Boolean = ...
2 val b : Boolean = a || !a
```

Quelle est la valeur de b? ☐ Elle dépend du contenu de a ☒ **true** ☐ false

[2 Bo] (d) Écrivez le code (sans fonction) permettant d'écrire tous les multiples de 79 plus grands que 1 et plus petits que 1000 sur la console.

Solution:

```
1 for (i: Int <- 1 until 1000; if i % 79 == 0)
2   println(i)
```

[4 Pt] (e) Vrai ou faux?

.....	True	False
(255+1).toByte == (0xFFFF+1).toShort	<input checked="" type="checkbox"/>	<input type="checkbox"/>
.....	True	False
Un générateur de séquence peut générer des Double	<input type="checkbox"/>	<input checked="" type="checkbox"/>
.....	True	False
0 to 10 est une séquence plus longue que 1 until 11	<input checked="" type="checkbox"/>	<input type="checkbox"/>
.....	True	False
\ est le caractère d'échappement	<input checked="" type="checkbox"/>	<input type="checkbox"/>
.....	True	False
Un programme qui compile fonctionne toujours correctement	<input type="checkbox"/>	<input checked="" type="checkbox"/>
.....	True	False
Le compilateur détecte les erreurs de type	<input checked="" type="checkbox"/>	<input type="checkbox"/>
.....	True	False
println() est une fonction qui retourne un String	<input type="checkbox"/>	<input checked="" type="checkbox"/>
.....	True	False
Dans l'expression if (foo) a else b, a et b doivent être de même type	<input type="checkbox"/>	<input checked="" type="checkbox"/>
.....		

Question 2 – Loops analysis (8 points)Que vont afficher **exactement** les boucles suivantes sur la console?

(a)

```

1  var foo: Int = 3
2  var bar: Int = 8
3  var i : Int = 8
4
5  do {
6      bar -= bar / (i>>1)
7      foo += 1 & 0
8      println(bar + foo + 1)
9  } while (foo + 1 < bar)

```

Solution :

Solution:

10
9
8

(b)

```

1  var j = 6
2  var i = 0
3  while (i != j) {
4      print(i + " " + j + " * ")
5      i += 1
6      j -= 1
7  }

```

Solution :

Solution:

0 6 * 1 5 * 2 4 *

(c)

```

1  var a: Int = 0xf0
2  var t: String = ""
3  while (a != 0) {
4      t = (if (a % 2 == 0) '0' else '1') + t
5      a /= 2
6  }
7  print(t)

```

Solution :

Solution:

11110000

(d)

```

1 // println(5)
2 for (i: Int <- 3 to 7){
3   print(s"${i-1*2/3} ")
4 }

```

Solution :

Solution:
3 4 5 6 7

Question 3 – EBNF grammars (6 points)

(a) Soit la grammaire suivante pour exp

```

factor ::= 'x' | 'y' | 'z' | parexpr
parexpr ::= '(' exp ')'
op ::= '+' | '-'
exp ::= (factor op exp) | factor

```

[1 Pt]

1) Donnez une production valide la plus courte possible pour exp.

Solution: x ou alors y ou alors z

[1 Pt]

2) Donnez une production utilisant chacune des règles de la grammaire.

Solution: x + (z-y)

[1 Pt]

3) Donnez une production de exp utilisant au moins deux fois la règle parexpr.

Solution: ((x + x) - (y + z))

[3 Pt]

(b) Écrivez la description EBNF de la grammaire *even-integer* qui reconnaît uniquement les entiers pairs. Par exemple, dans cette grammaire -6 et 34 sont valides alors que 3 et -23 ne le sont pas. On considère également que 0 et -0 sont valides.

Solution:

```

sign ::= '+' | '-'
even-digit ::= '0' | '2' | '4' | '6' | '8'
digit ::= even-digit | '1' | '3' | '5' | '7' | '9'
even-integer ::= [sign] {digit} even-digit

```

Question 4 – Code comprehension (3 points)

Analysez la fonction suivante puis répondez aux questions ci-dessous.

```

1  def isL(x: Char): Boolean = {
2      if (x >= 'a' && x <= 'z') true else false
3  }
4
5  def isU(x: Char): Boolean = {
6      if (x >= 'A' && x <= 'Z') true else false
7  }
8
9  def le(x: Char): Boolean = {
10     isU(x) || isL(x)
11 }
12
13 def bar(word: String): Int = {
14     var p: Int = 0
15
16     for (i <- word) {
17         if (!le(i))
18             return p
19
20         p += 1
21     }
22     return -1
23 }

```

- [2 Pt] (a) Expliquez avec des phrases à quoi sert la fonction bar ci-dessus.

Solution: La fonction sert à trouver la position du premier caractère de la chaîne qui n'est pas une lettre minuscule ou majuscule, par exemple un chiffre ou un espace par exemple. Si la chaîne est composée uniquement de lettres ou elle est vide, la fonction retourne -1.

- [1 Pt] (b) Donnez **deux exemples** complets d'utilisation de bar permettant de démontrer votre explication.

Solution: Exemple 1: bar("Hello World") retourne 5 et bar("foo") retourne -1.

Question 5 – Writing functions (7 points)

- [2 Pt] (a) Écrivez une fonction nommée foo qui prend un Double nommé x en argument et qui retourne la valeur

$$foo(x) = \frac{3x^2 + \sin(x) - 3}{x^3}$$

Solution:

```

1  def foo(x: Double) : Double = {
2      return (3*x*x + math.sin(x) - 3) / (x*x*x)
3  }

```

- [2 Pt] (b) Écrivez une fonction retournant, à partir d'un nombre de jours entiers, le nombre correspondant de secondes.

Solution:

```

1  def nSeconds(days: Int) : Int = {
2      return days * 60 * 60 * 24
3  }

```

- [3 Pt] (c) 1) Écrivez une fonction nommée `factorFinder` qui affiche sur la console tous les diviseurs entiers (sans lui-même ni 1) d'un nombre entier passé en argument. Si le nombre n'a pas de diviseur (à part lui-même et 1 s'entend), affichez que c'est un nombre premier. **Par simplification, on considère que la fonction ne recevra que des nombres ≥ 3 .** Exemple :

```

1 factorFinder(12) -> affiche "The whole dividers of 12 are : 6 4 3 2 "
2 factorFinder(54) -> affiche "The whole dividers of 54 are : 27 18 9 6 3 2 "
3 factorFinder(51) -> affiche "The whole dividers of 51 are : 17 3 "
4 factorFinder(3)  -> affiche "The whole dividers of 3 are : 3 is prime !"
5 factorFinder(541) -> affiche "The whole dividers of 541 are : 541 is prime !"

```

Solution:

```

1 def factorFinder(n: Int): Unit = {
2   print(s"The whole dividers of $n are : ")
3
4   var nDividers = 0
5
6   for (i <- n - 1 until 1 by -1) {
7     if (n % i == 0) {
8       print(s"$i ")
9       nDividers +=1
10    }
11  }
12
13  if(nDividers == 0)
14    print(s"$n is prime !")
15 }

```

- [1 Bo] 2) Dans le code ci-dessus, on constate que le string affiché contient un espace à la fin. Comment pouvez-vous faire pour effacer ce caractère dans la console s'il a déjà été généré ?

Solution: Avec l'aide du caractère spécial `\b`

Question 6 – String manipulations (11 points)

Considérez que vous avez à disposition dans votre code les fonctions suivantes pour manipuler les chaînes de caractères, **et uniquement celles-ci**.

```
1  /** Returns the length (number of letters) of the String s */
2  def length(s: String) : Int
3  /** Returns the char at position pos in s */
4  def charAt(s: String, pos: Int)
```

[3 Pt] (a) **Progressive strings**

00h	␣ (nul)	10h	► (dle)	20h	□	30h	0	40h	@	50h	P	60h	'	70h	p
01h	Ⓢ (soh)	11h	◄ (dcl)	21h	!	31h	1	41h	A	51h	Q	61h	a	71h	q
02h	Ⓢ (stx)	12h	‡ (dc2)	22h	"	32h	2	42h	B	52h	R	62h	b	72h	r
03h	Ⓢ (etx)	13h	‡ (dc3)	23h	#	33h	3	43h	C	53h	S	63h	c	73h	s
04h	Ⓢ (eot)	14h	‡ (dc4)	24h	\$	34h	4	44h	D	54h	T	64h	d	74h	t
05h	Ⓢ (enq)	15h	§ (nak)	25h	%	35h	5	45h	E	55h	U	65h	e	75h	u
06h	Ⓢ (ack)	16h	— (syn)	26h	&	36h	6	46h	F	56h	V	66h	f	76h	v
07h	Ⓢ (bel)	17h	‡ (etb)	27h	'	37h	7	47h	G	57h	W	67h	g	77h	w
08h	Ⓢ (bs)	18h	† (can)	28h	(38h	8	48h	H	58h	X	68h	h	78h	x
09h	Ⓢ (tab)	19h	↓ (em)	29h)	39h	9	49h	I	59h	Y	69h	i	79h	y
0Ah	Ⓢ (lf)	1Ah	(eof)	2Ah	*	3Ah	:	4Ah	J	5Ah	Z	6Ah	j	7Ah	z
0Bh	Ⓢ (vt)	1Bh	← (esc)	2Bh	+	3Bh	;	4Bh	K	5Bh	[6Bh	k	7Bh	{
0Ch	Ⓢ (np)	1Ch	L (fs)	2Ch	,	3Ch	<	4Ch	L	5Ch	\	6Ch	l	7Ch	
0Dh	Ⓢ (cr)	1Dh	↔ (gs)	2Dh	-	3Dh	=	4Dh	M	5Dh]	6Dh	m	7Dh	}
0Eh	Ⓢ (so)	1Eh	▲ (rs)	2Eh	.	3Eh	>	4Eh	N	5Eh	^	6Eh	n	7Eh	~
0Fh	Ⓢ (si)	1Fh	▼ (us)	2Fh	/	3Fh	?	4Fh	O	5Fh	_	6Fh	o	7Fh	␣

FIGURE 1 – La table ASCII

Écrivez une fonction qui reçoit un String en argument (de taille 1 au minimum) et qui vérifie que les lettres dans le String sont toutes classées dans l'ordre de la table ASCII. Exemple :

```
1  progressive("ABCD") // Returns true
2  progressive("Aa") // Returns true
3  progressive("04Tu") // Returns true
4  progressive("ba") // Returns false
5  progressive("cZ") // Returns false
```

Solution:

```
1  def progressive(s: String): Boolean = {
2      var previous : Char = s.charAt(0)
3
4      for(i <- 1 until s.length){
5          if(previous > s.charAt(i)) {
6              return false
7          }
8
9          previous = s.charAt(i)
10     }
11     return true
12 }
```

[4 Pt] (b) Double vowels

Écrivez la fonction `doubleVowels` qui reçoit un `String` en argument et retourne un `String`. Le `String` retourné correspond au `String` reçu mais avec toutes les voyelles qui ont été doublées. Pour cet exercice les voyelles sont : a, e, i, o, u, y.

On considère par simplification que la chaîne en entrée est toujours en minuscule (donc pas besoin de gérer les lettres majuscules). Exemple :

```
1 doubleVowels("hello") // Returns "heelloo"
2 doubleVowels("aabb") // Returns "aaaabb"
```

Solution:

```
1 def doubleVowels(s:String) : String = {
2   var t: String = ""
3   for (c <- s) {
4     c match {
5       case 'a'|'e'|'i'|'o'|'u'|'y' =>
6         t = t + c + c
7       case _ =>
8         t += c
9     }
10  }
11  return t
12 }

13 // Other solution
14 def doubleVowels(s:String) : String = {
15   var t: String = ""
16   for (c <- s) {
17     t += c match {
18       case 'a'|'e'|'i'|'o'|'u'|'y' =>
19         c + "" + c
20       case _ =>
21         c
22     }
23   }
24   return t
25 }
26 }
```

[4 Pt] (c) No triples

Le clavier de votre ordinateur est défectueux : quand vous tapez 2x de suite la même touche, il écrit 3x de suite le même caractère !

Écrivez la fonction `noTriples` qui corrige cette erreur dans la chaîne passée en argument. Notez que cette fonction n'interagit pas avec la console. Exemple :

```
1 "J'ai fait cettte illlusion" devient "J'ai fait cette illusion"
```

Solution:


```
1  def noTriples(s:String) : String = {  
2    var t: String = ""  
3    var last: Char = 0  
4    var count: Int = 0  
5  
6    for (c <- s) {  
7      if (last == c) {  
8        count += 1  
9      } else {  
10       count = 0  
11     }  
12  
13     if (count != 2) {  
14       t += c  
15     }  
16  
17     last = c  
18   }  
19   return t  
20 }
```

Question 7 – Too fast, too furious (5 points)

Afin d'améliorer la sécurité sur les routes, les pandores valaisans vont installer un nouveau radar ultra moderne. Ce radar est en effet capable de calculer la vitesse moyenne entre deux points séparés par une certaine distance.

Une infraction est alors constatée si la vitesse moyenne du véhicule est supérieure à la limitation de vitesse en cours.

Écrivez les fonctions suivantes :

- `def averageSpeed(timeOne : Double, timeTwo : Double, distance : Double)` retournant la vitesse moyenne entre les deux radars. Les deux paramètres de temps sont en secondes (remarque : seule la différence entre les deux temps est importante) et la valeur de retour doit être en km/h. La distance entre les deux radars est donnée en mètres.
- `isFaster`, un prédicat prenant comme arguments la vitesse du chauffeur et la limitation de vitesse et qui retourne si le chauffeur est en dehors des limitations.
- `getFineAmount(driverSpeed : Double, maxSpeed : Int)` retournant la valeur de l'amende que le chauffeur devra payer en cas d'infraction.

Pour le dernier point, le montant de l'amende est calculé comme suit :

- une déduction de 5 % doit être faite sur la vitesse mesurée. Arrondissez le résultat de cette déduction à l'unité (ex. 3.4 donne 3 et 3.6 donne 4).
- la facture s'élèvera à 60.- par tranche entamée de 10 km/h supérieurs à la limitation. Ainsi, le montant de l'amende doit être arrondi à la dizaine supérieure (23 km/h en trop vont être facturés comme 30 km/h).

Votre solution :

Solution:

```
1  def averageSpeed(time1: Double, time2: Double, distance: Double) = (distance / (time2
   - time1)) * 3.6
2  def isFaster(speed: Double, maxSpeed: Double) = speed > maxSpeed
3
4  def getFineAmountSol1(speed: Double, maxSpeed: Int) = {
5      val realSpeed = ((speed - speed * 5.0 / 100.0) + 0.5).toInt
6      var price = 0
7      var s = realSpeed - maxSpeed
8      while (s > 0) {
9          price += 60
10         s -= 10
11     }
12     price
13 }
14
15 // Other solution
16 def getFineAmountSol2(speed: Double, maxSpeed: Int) = {
17     val realSpeed = ((speed - speed * 5.0 / 100.0) + 0.5).toInt
18     val diff = realSpeed - maxSpeed
19
20     if (diff % 10 == 0)
21         diff / 10 * 60
22     else if (diff > 0)
23         ((diff / 10) + 1) * 60
24     else 0
25 }
```

Fin|Ende
