

Les documents sont interdits.

Ce contrôle contient un questionnaire à choix multiples qui sera corrigé automatiquement. Il est donc important de remplir avec soin les réponses au stylo noir (le crayon de papier ou les stylos plus clairs peuvent poser des problèmes de lecture). Lorsque vous choisissez une réponse, il faut noircir complètement la case correspondante. Ne pas toucher aux cases situées tout en haut de la feuille, elles servent à repérer vos copies lors de la correction automatisée.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Les questions ♣ sont notées sur 2 points, -1 point par mauvaise réponse (0 minimum par question). Les questions à choix unique sont notées sur 2 points.

- 0 0 0 0 0 0 0 0
- 1 1 1 1 1 1 1 1
- 2 2 2 2 2 2 2 2
- 3 3 3 3 3 3 3 3
- 4 4 4 4 4 4 4 4
- 5 5 5 5 5 5 5 5
- 6 6 6 6 6 6 6 6
- 7 7 7 7 7 7 7 7
- 8 8 8 8 8 8 8 8
- 9 9 9 9 9 9 9 9

← codez votre numéro d'étudiant ci-contre, et écrivez votre nom et prénom ci-dessous.

Nom et prénom :

.....

1 λ-calcul: syntaxe

On rappelle que le λ-calcul est l'ensemble de formules Λ défini par la grammaire suivante où Var est un ensemble de variables et E, F ∈ Λ :

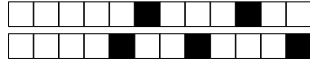
$$\Lambda ::= x \in \text{Var} \mid (E F) \mid \lambda x.E$$

Question 1 ♣ Parmi les expressions suivantes, indiquer celles qui sont des expressions syntaxiquement valides du λ-calcul:

- ((x y) z)
- ((λx.(x y))z)
- (λz.(λy.(y z)))(λx)
- Aucune de ces réponses n'est correcte.

Question 2 Indiquer quel est l'ensemble des variables libres de l'expression (λx.(λz.y))(λy.y) :

- {x, y}
- {x}
- ∅
- {y}



Question 3 ♣ Parmi les expressions suivantes, indiquer celles qui sont des expressions syntaxiquement valides du λ -calcul:

- $\lambda y.(\lambda x.x)$ $\lambda x.(\lambda x.x)$
 $\lambda z.(\lambda x.y)$ *Aucune de ces réponses n'est correcte.*

Question 4 Indiquer quel est l'ensemble des variables libres de l'expression $\lambda x.(\lambda z.y)$:

- $\{x, y\}$ $\{x\}$
 $\{y\}$ \emptyset

Question 5 Indiquer quel est l'ensemble des variables libres de l'expression $\lambda x.(\lambda y.x)$:

- $\{y\}$ $\{x\}$
 $\{x, y\}$ \emptyset

2 λ -calcul: β -réduction

On considère dans cette partie et dans la suite que le λ -calcul intègre les entiers naturels et les opérations $*$ et $+$ ayant les types et réductions évidentes. Ce langage est défini par la grammaire suivante :

$$\Lambda := n \in \mathbf{N} \mid x \in \text{Var} \mid (E F) \mid \lambda x.E \mid E + F \mid E * F$$

Question 6 ♣ Parmi les expressions suivantes, indiquer celles dans lesquelles l'expression $((\lambda x.((\lambda y.y * x) x)) 5)$ se β -réduit en *une* ou en *plusieurs* étapes :

- 25 $((\lambda y.(y * 5)) 5)$
 $\lambda x.((5 * x) x)$ *Aucune de ces réponses n'est correcte.*

Question 7 ♣ Parmi les expressions suivantes, indiquer celles dans lesquelles l'expression $((\lambda x.x + 5)((\lambda z.z) 2))$ se β -réduit en *une* ou en *plusieurs* étapes :

- 7 $(\lambda x.x + 5) 2$
 $((\lambda z.z) 2) + 5$ *Aucune de ces réponses n'est correcte.*

Question 8 ♣ Parmi les expressions suivantes, indiquer celles dans lesquelles l'expression $((\lambda x.(x x))(\lambda x.(x x)))$ se β -réduit en *une* ou en *plusieurs* étapes :

- $((\lambda x.(x x))(\lambda x.(x x)))$ $(\lambda x.(x x))$
 $(\lambda x.(\lambda x.(x x))(\lambda x.(x x)))$ *Aucune de ces réponses n'est correcte.*

Question 9 ♣ Parmi les expressions suivantes, indiquer celles dans lesquelles l'expression $((\lambda x.x)(\lambda y.y)) 3$ se β -réduit en *une* ou en *plusieurs* étapes :

- 3 $((\lambda y.y) 3)$
 $((\lambda x.(\lambda y.y)) 3)$ *Aucune de ces réponses n'est correcte.*



Question 10 Parmi les expressions suivantes, indiquer celle dans laquelle l'expression $(((\lambda x.(\lambda y.(y(3+x)))) 4) (\lambda z.(z * 2)))$ se β -réduit en *une* ou en *plusieurs* étapes :

- L'expression ne se réduit pas en un entier. 14
 11 10

3 λ -calcul: typage

Question 11 On considère que le seul type primitif du λ -calcul est `number`. Indiquer quel est le type de l'expression $(\lambda f.\lambda x.(f(f x)))(\lambda z.z + 2)$:

- L'expression n'est pas typable. `number \rightarrow (number \rightarrow number)`
 `(number \rightarrow number) \rightarrow number` `number \rightarrow number`

Question 12 On considère que le seul type primitif du λ -calcul est `number`. Indiquer quel est le type de l'expression $(\lambda x.\lambda y.x)(\lambda z.z + 2)$:

- `number \rightarrow (number \rightarrow number)` `number \rightarrow number`
 `(number \rightarrow number) \rightarrow number` L'expression n'est pas typable.

Question 13 ♣ On considère que le seul type primitif du λ -calcul est `number`. Parmi les expressions suivantes, indiquer celles qui ont pour type `(number \rightarrow number) \rightarrow (number \rightarrow number)` :

- $\lambda f.\lambda x.(f x)$ $\lambda f.(f 2)$
 $\lambda f.\lambda x.(f(f x))$ Aucune de ces réponses n'est correcte.

Question 14 ♣ On considère que le seul type primitif du λ -calcul est `number`. Parmi les expressions suivantes, indiquer celles qui sont correctement typables :

- $(\lambda x.(x + 3))(\lambda y.y)$ $\lambda f.(f f)$
 $(((\lambda x.(\lambda y.(x y))) 3) 4)$ Aucune de ces réponses n'est correcte.

Question 15 On considère que le seul type primitif du λ -calcul est `number`. Indiquer quel est le type de l'expression $(\lambda f.(f 2) + (f 3))$:

- `(number \rightarrow number) \rightarrow number` `number \rightarrow number`
 L'expression n'est pas typable. `number \rightarrow (number \rightarrow number)`



4 λ -calcul et évaluation du javascript

On rappelle que l'application en javascript se note différemment du λ -calcul, on écrit en javascript $f \Rightarrow x \Rightarrow f(x)$ pour l'équivalent $\lambda f.\lambda x.(f x)$ du λ -calcul.

Question 16 ♣ Soit la fonction `let fn3 = x => x(x)`. Parmi les programmes suivants, indiquer lesquels produisent la valeur 3

- `fn3(x => x)(3)` `fn3(3)`
 `fn3(fn3(x => x))(3)` Aucune de ces réponses n'est correcte.

Question 17 ♣ Soit les fonctions `let I = x => x`; et `let K = x => y => x`; . Parmi les programmes suivants, indiquer lesquels produisent la valeur 3

- `K(K)(1)(3)(2)` `K(K(3))(1)(2)`
 `K(K(1))(3)(2)` Aucune de ces réponses n'est correcte.

Question 18 On considère la définition javascript `let ex2 = f => g => x => f(g(x))`; , indiquer quelle expression du λ -calcul la représente parmi les suivantes :

- $\lambda f.\lambda g.\lambda x.(f (g x))$ $\lambda f.\lambda g.(f (g x))$
 $\lambda f.\lambda g.((f g) x)$ $\lambda f.\lambda g.\lambda x.((f g) x)$

Question 19 ♣ Soit les fonctions `let I = x => x`; et `let K = x => y => x`; . Parmi les programmes suivants, indiquer lesquels produisent la valeur 3

- `K(3)(1)` `K(I)(1)(3)`
 `K(I)(3)(1)` Aucune de ces réponses n'est correcte.

Question 20 On considère la définition javascript `let plus = x => y => x + y`; quel est le résultat de l'évaluation de `(x => plus(2))(x)(3)` :

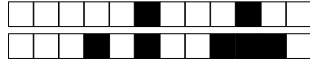
- la valeur 5 ReferenceError: x is not defined
 NaN (pour Not a Number) function ()

Question 21 On considère les définitions javascript `let comp = f => g => x => f(g(x))`; et `let ex3 = comp(x => x+2)(y => y*3)`; indiquer quelle définition est équivalente parmi les suivantes :

- `let ex3= x => x*3 + 2;` `let ex3= x => (x*2)+ 3;`
 `let ex3= x => (x+2)* 3;` `let ex3= x => (y => y*3)((z => z+2)(x));`

Question 22 On considère la définition javascript `let plus = x => y => x + y`; quel est le résultat de l'évaluation de `(x => plus(2))(x)(3)` :

- la valeur 5 function ()
 ReferenceError: x is not defined NaN (pour Not a Number)



Question 23 ♣ Soit les fonctions `let I = x => x`; et `let K = x => y => x`;. Parmi les programmes suivants, indiquer lesquels produisent la valeur 3

`I(K(I))(3)(1)`

`I(I)(3)`

`I(K)(3)`

Aucune de ces réponses n'est correcte.

Question 24 On considère l'expression $\lambda f.\lambda x.(f (f x))$, indiquer quelle définition javascript la représente parmi les suivantes :

`let ex1 = f => x => f(f(x));`

`let ex1 = f => x => f (f x);`

`let ex1 = (f,x)=> f(f(x));`

`let ex1 = f => f(f(x));`

5 javascript

```
1 function fun0(n){
2   if (n===0)
3     return 0;
4   else
5     return (n + fun0(n-1));
6 }
7
8 function fun1(n){
9   let arr = [1];
10  for (let i = 1; i < n; i++)
11    arr.push(arr[i-1] * i);
12  return arr;
13 }
14
15 function fun2(a, b, c) {
16   let res = [];
17   if (typeof b === 'undefined')
18     b = 0;
19   if (typeof c === 'undefined')
20     c = 1;
21
22   for(let i = b; (i < a) ; i = i + c)
23     res.push(i);
24
25   return res;
26 }
```

Question 25 ♣ Soient `var sExample = '1'`; `var iExample = 1`; `var bExample = true`;. Parmi les expressions suivantes, indiquer celles qui s'évaluent à `true`

`iExample === bExample;`

`iExample == bExample;`

`sExample == bExample;`

Aucune de ces réponses n'est correcte.

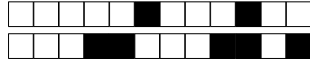
Question 26 ♣ Soient `var oExample0 = a : 0`; `var oExample1 = a : 0`;. Parmi les expressions suivantes, indiquer celles qui s'évaluent à `true`

`oExample0 == oExample1;`

`oExample0 == oExample0;`

`oExample0 === oExample0;`

Aucune de ces réponses n'est correcte.



Question 27 ♣ Parmi les expressions suivantes, indiquer celles qui s'évaluent à `true`; où `non_decl` est une variable déclarée nulle part.

- `typeof undefined === typeof non_decl;` `typeof "toto"=== typeof "string";`
 `typeof ""=== typeof [];` *Aucune de ces réponses n'est correcte.*

Question 28 ♣ On considère la fonction `fun0` définie dans l'encadré. Indiquer parmi les propositions suivantes celles qui sont correctes :

- `f(n) === 15;` `f(n) >= n; ∀n ≥ 0`
 `f(n) === (n => (n+1)*(n)/2)(n); ∀n ≥ 0` *Aucune de ces réponses n'est correcte.*

Question 29 ♣ On considère la fonction `fun1` définie dans l'encadré. Indiquer parmi les propositions suivantes celles qui sont correctes :

- `fun1(0) === fun1(1)` `fun1(undefined)=== []`
 `fun1(5) === [1, 1, 2, 6, 24]` *Aucune de ces réponses n'est correcte.*

Question 30 ♣ On considère la fonction `fun2` définie dans l'encadré. Indiquer parmi les propositions suivantes celles qui sont correctes :

- `fun2(5,10)=== [5, 6, 7, 8, 9]` `fun2(5,undefined,2)=== [0, 2, 4]`
 `fun2(5) === fun2(5,undefined,undefined)` *Aucune de ces réponses n'est correcte.*

Question 31 ♣ On considère la fonction `fun2` définie dans l'encadré. Indiquer parmi les propositions suivantes celles qui sont correctes :

- `fun2(undefined)=== undefined` `fun2(5,0,0)` tourne en boucle à l'infini
 `fun2(-10,-5)=== [-10, -9, -8, -7, -6]` *Aucune de ces réponses n'est correcte.*

Question 32 ♣ Soient `var tExample0 = [0]; var tExample1 = [0];`. Parmi les expressions suivantes, indiquer celles qui s'évaluent à `true`

- `tExample0[0] == tExample1[0];` `tExample0 == tExample1;`
 `oExample0 == [0];` *Aucune de ces réponses n'est correcte.*