

Correction du Partiel TYLA

TPOLOGIE DES LANGAGES

EPITA 2013 – Sans document ni machine

Juin 2011 (1h00)

Correction: Le sujet et sa correction ont été écrits par Roland Levillain et Akim Demaille.

Barème: Se reporter à la feuille de calcul pour les coefficients des questions.

Répondre sur les formulaires de QCM; aucune réponse manuscrite ne sera corrigée. Renseigner les champs d'identité. Bien lire les questions, chaque mot est important. Il y a une seule réponse juste pour ces questions. Lorsque plusieurs sont valides, sélectionner la plus restrictive. Par exemple s'il est demandé si 0 est *nul*, *non nul*, *positif*, ou *négatif*, cocher *nul* qui est plus restrictif que *positif* et *négatif*, tous deux vrais. Répondre incorrectement est plus pénalisé que de ne pas répondre.

1 Programmation orientée objet

Q.1 Le type dynamique d'un objet

- ✓ est un sous-type de son type statique.
- ✗ est un sur-type de son type statique.
- ✗ est connu à la compilation.
- ✗ est utilisé pour distinguer des fonctions/méthodes surchargées.

Q.2 Dans quel langage les appels de méthodes ne sont pas vérifiés statiquement?

- ✗ C++
- ✗ C#
- ✗ Java
- ✗ Simula
- ✓ Smalltalk

Q.3 Qu'appelle-t-on une métaclasse en Smalltalk ?

- ✗ Une classe abstraite.
- ✗ Une classe qui hérite d'elle-même.
- ✗ Une classe ayant des méta-méthodes.
- ✓ Une classe dont les instances sont des classes.

Q.4 Les multiméthodes permettent

- ✗ aux méthodes de retourner plusieurs résultats.
- ✓ le polymorphisme dynamique sur plusieurs arguments de fonctions.
- ✗ à une classe d'avoir des méthodes portant le même nom.
- ✗ d'avoir des méthodes polymorphes (virtuelles) dans une hiérarchie de classe utilisant l'héritage multiple.

2 Programmation générique

Q.5 Les templates de classes du C++ sont

- ☒ des collections de templates de fonctions libres.
- ☒ des générateurs de classes.
- ☒ des classes dont toutes les méthodes sont virtuelles.
- ☒ des classes dont toutes les méthodes sont virtuelles pures.

Q.6 Parmi les termes suivants, lequel ne peut pas être utilisé comme paramètre effectif d'une classe paramétrée ?

- ☒ `const`.
- ☒ Une constante entière.
- ☒ `unsigned`.
- ☒ Un type classe défini par l'utilisateur.

Q.7 Parmi les lignes C++ suivantes, laquelle est invalide ?

- ☒ `std::pair p1 (42, 51);`
- ☒ `std::pair<float, int> p2 (42, 3.14f);`
- ☒ `std::pair<int, float> p3 = std::make_pair (42, 3.14f);`
- ☒ `std::pair<int, float> p4 = std::make_pair<int, float> (2.72f, 51);`

3 Programmation fonctionnelle

Q.8 On dit d'un langage qu'il est fonctionnel si...

- ☒ il n'effectue aucun effet de bord.
- ☒ il permet de manipuler des fonctions comme n'importe quel autre entité/objet.
- ☒ il supporte le concept de fonction récursive.
- ☒ il est Turing complet.
- ☒ il dispose d'un compilateur implémenté et en état de marche.

Q.9 Un langage fonctionnel est dit pur lorsque

- ☒ il proscrie tout effet de bord.
- ☒ il ne contient aucune construction orientée objet.
- ☒ ses fonctions ont au plus un argument.
- ☒ ses expressions sont évaluées paresseusement.
- ☒ la récursion est proscrite.

Q.10 On appelle fermeture

- ☒ une fonction qui n'est pas récursive.
- ☒ une fonction qui capture des références à des variables libres dans l'environnement lexical.
- ☒ une fonction qui a été mise en ligne (*inlined*).
- ☒ une fonction passée en argument à une autre fonction.

4 C++

Q.11 En C++, on appelle objet-fonction

- ☒ un objet construit à l'intérieur d'une fonction.
- ☒ un objet disposant d'un `operator()`.
- ☒ une méthode.
- ☒ un fichier de code compilé ('foo.o') ne contenant qu'une seule fonction (`foo()`).

Q.12 La liaison dynamique en C++

- ☒ a un rapport avec `virtual`.
- ☒ est liée à la surcharge des opérateurs.
- ☒ repose sur `template`.
- ☒ s'appuie sur `dynamic_cast`.

Q.13 Surcharge vs méthodes virtuelles: quelle est la bonne réponse ?

- ✗ La surcharge et les méthodes virtuelles sont des mécanismes dynamiques.
- ✗ La surcharge et les méthodes virtuelles sont des mécanismes statiques.
- ✓ La surcharge est un mécanisme statique, les méthodes virtuelles un mécanisme dynamique.
- ✗ La surcharge est un mécanisme dynamique, les méthodes virtuelles un mécanisme statique.

Q.14 Lequel de ces éléments n'entre pas en compte lors de la résolution d'une méthode surchargée en C++?

- ✗ les arguments de l'appel.
- ✗ le qualificatif const de la méthode.
- ✗ l'arité de la fonction.
- ✓ le type de retour.
- ✗ le nom de la fonction.

5 Langages de programmation

Q.15 Qui est l'auteur du langage C?

- ✗ Brian Kernighan ✗ Bjarne Stroustrup ✗ K. N. King
- ✓ Dennis Ritchie
- ✗ Ken Thompson

Q.16 Quelle société est à l'origine des systèmes de fenêtrage, de la souris, de l'imprimante laser?

- ✗ Apple
- ✗ Apollo
- ✗ IBM
- ✗ Microsoft
- ✓ Xerox

Q.17 Lequel de ces langages n'a pas été influencé par Simula ?

- ✓ Algol
- ✗ C++
- ✗ Objective C
- ✗ Smalltalk
- ✗ Eiffel

Q.18 Qui est l'inventeur de la souris ?

- ✗ Walt Disney
- ✗ Bart
- ✗ Gordon Moore
- ✓ Douglas Engel
- ✗ Donald Knuth
- ✗ Konrad Zuse

Q.19 Que signifie "BNF" ?

- ✓ Backus-Naur Form
- Formalism
- mat
- (automaton)
- ✗ BASIC
- Numbering
- ✗ Bison
- Normal
- For
- ✗ Bound
- Non-Finite

6 Fonctions

Q.20 Le support des fonctions récursives nécessite

- ✗ un tas (*heap*).
- ✓ une pile (*stack*).
- ✗ la liaison des fonctions dynamiques.
- ✗ que le langage dispose de pré-déclarations (*forward declarations*).

Q.21 À la fin de ce programme, avec un *Mode* de passage des arguments par copie, quelles sont les valeurs des l-values?

```
var t      : integer
    foo    : array [1..2] of integer;

procedure shoot_my(x : Mode integer);
begin
    foo[1] := 6;
    t      := 2;
    x      := x + 3;
end;
```

```
begin
    foo[1] := 1;
    foo[2] := 2;
    t      := 1;
    shoot_my (foo[t]);
end.
```

✓ foo[1] = 6, foo[2] = 2, t = 2

✗ foo[1] = 6, foo[2] = 4, t = 2

✗ foo[1] = 9, foo[2] = 2, t = 2

✗ foo[1] = 6, foo[2] = 5, t = 2

Q.22 Même question, mais avec un *Mode* de passage d'arguments par référence.

✗ foo[1] = 6, foo[2] = 2, t = 2

✗ foo[1] = 6, foo[2] = 4, t = 2

✓ foo[1] = 9, foo[2] = 2, t = 2

✗ foo[1] = 6, foo[2] = 5, t = 2