



UE INF1601

2020

Théorie des langages et compilation
Contrôle continu numéro 1
(20 minutes)

Nom et prénom :

Peter Budovic

Noircissez les bonnes réponses (cocher ne suffit pas). Les questions faisant apparaître le symbole ♣ peuvent présenter une ou plusieurs bonnes réponses ; les autres ont une seule bonne réponse. Toute absence de réponse équivaut à une réponse fausse. Utilisez le verso des feuilles comme brouillon si nécessaire.

Rôle d'un compilateur

Question 1 ♣ Quel est la traduction anglaise de compilateur ?

- ☒ language processor
☐ compiler

☒ compiler
☐ parser

☐ assembler
☐ word processor

Question 2 ♣ Parmi les langages suivants, lesquels sont de bas niveau ?

- ☒ le code machine
☐ le shell Unix
☒ le langage d'assemblage

☐ le langage C
☒ un langage dans lequel chaque instruction correspond à une instruction machine

Question 3 Un assembleur est un traducteur.

- ☐ faux
 ☒ vrai

Question 4 Un processeur est un interpréteur.

- ☐ faux
 ☒ vrai

Question 5 Un compilateur traduit chaque instruction du programme source par une seule instruction machine.

- ☒ faux
 ☐ vrai

Question 6 Un interpréteur traduit la totalité du programme source avant de commencer à l'exécuter.

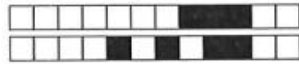
- ☒ faux
 ☐ vrai

Question 7 Un compilateur traduit un programme source vers un langage cible exécutable.

- ☒ vrai
 ☐ faux

Question 8 Une machine virtuelle est un interpréteur.

- ☒ vrai
 ☐ faux



Question 9 Le programme source d'un langage semi-interprété est traduit partiellement avant d'être interprété.

☐ vrai

☒ faux

Question 10 Le programme source d'un langage compilé est traduit complètement avant d'être exécuté.

☐ faux

☒ vrai

Question 11 Le pré-traitement effectué par un interpréteur est plus important que celui réalisé par un compilateur.

☐ vrai

☒ faux

Question 12 La compilation à la volée (JIT) dans une JVM consiste à traduire le programme source Java en code binaire à l'exécution.

☒ faux

☐ vrai

Question 13 Les langages semi-interprétés s'exécutent plus vite que les langages compilés.

☒ faux

☐ vrai

Question 14 ♣ On attend d'un compilateur :

☒ qu'il produise du code compact

☐ qu'il corrige les erreurs de programmation

☐ qu'il exécute le programme s'il ne comporte pas d'erreur

☒ qu'il produise du code efficace

☐ qu'il rejette les programmes contenant trop de variables (> 1000)

Question 15 ♣ Parmi les outils suivants, lesquels sont des traducteurs ?

☒ Un éditeur de liens

☒ Un assembleur

☒ Un interpréteur

☒ Une machine virtuelle

☐ Une mémoire virtuelle

☒ Un compilateur

Etape d'une compilation

Question 16 Un programme source écrit dans un langage de programmation est un fichier au format texte.

☐ faux

☒ vrai

Question 17 L'analyse lexicale produit un arbre.

☒ faux

☐ vrai

Question 18 L'analyse syntaxique produit un arbre et une table.

☒ vrai

☐ faux

Question 19 AST est l'acronyme de Arbre Syntaxique Abstrait

☐ faux

☒ vrai



Question 20 L'analyse sémantique corrige les erreurs de type.

☒ vrai

☒ faux

Question 21 L'analyseur sémantique produit du code intermédiaire.

☐ vrai

☒ faux

Question 22 Le générateur de code intermédiaire prend en entrée un arbre.

☒ faux

☒ vrai

Méthode de conception

Question 23 Le paradigme analyse-synthèse est un découpage fonctionnel du compilateur.

☐ faux

☒ vrai

Question 24 Les langages formels sont plus expressifs que les langages naturels.

☒ faux

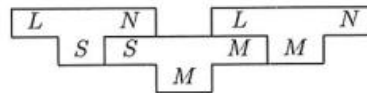
☐ vrai

Question 25 Le formalisme BNF permet de décrire la grammaire d'un langage de programmation.

☒ vrai

☐ faux

Question 26 Le schéma suivant illustre une compilation croisée de L pour N sur M .



☒ vrai

☒ faux

Question 27 La compilation croisée permet de produire un compilateur qui s'exécute sur une autre machine.

☒ faux

☐ vrai

Question 28 Le premier compilateur a été écrit en assembleur pour le langage Fortran.

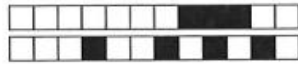
☐ faux

☒ vrai

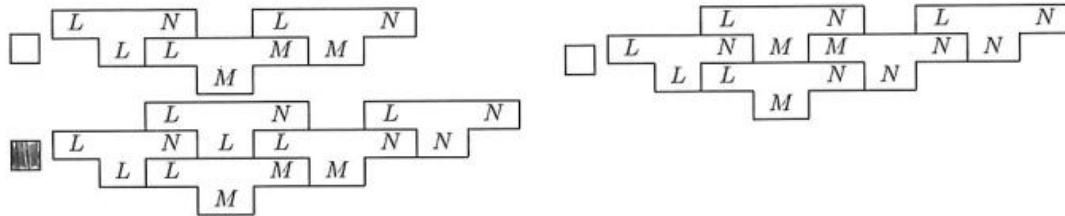
Question 29 L'auto-compilation permet d'écrire un compilateur dans le langage qu'il compile.

☒ vrai

☐ faux



Question 30 On dispose d'un compilateur d'un langage L écrit en L produisant du code pour une machine M . Et on dispose du compilateur du langage L sur M . On réécrit le backend du compilateur de L pour une machine N et on obtient un compilateur de L sur une machine N de la manière suivante :



1.5/1.5

Structure d'un compilateur

Question 31 La partie du compilateur avant réalise l'analyse du programme source.

☐ faux☒ vrai

Question 32 Le back-end du compilateur produit le programme cible.

☐ faux☒ vrai

Question 33 Le paradigme analyse-synthèse facilite l'exécution du compilateur.

☒ faux☐ vrai

Question 34 Pour recibler un compilateur il faut réécrire seulement sa partie arrière.

☐ faux☒ vrai

Question 35 ♣ Quelles étapes contient la partie avant du compilateur (l'ordre donné ci-dessous ne doit pas être pris en compte) ?

☒ l'analyse syntaxique☒ l'analyse lexicale☒ la génération de code intermédiaire☒ l'optimisation sémantique

Question 36 ♣ Quelles étapes contient la partie arrière du compilateur (l'ordre donné ci-dessous ne doit pas être pris en compte) ?

☒ optimisation du code cible☐ l'analyse syntaxique☒ la génération de code cible☐ l'analyse lexicale

Question 37 ♣ Dans quelle partie est utilisée l'arbre syntaxique abstrait ?

☐ Dans la partie arrière☒ Dans la partie avant

Le langage PTS

Question 38 Le langage PTS est un langage semi-interprété.

☒ faux☒ vrai

Question 39 Le langage PTS est un langage impératif.

☐ faux☒ vrai

1/1

1/1

1/1

1/1

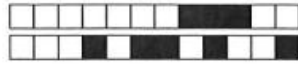
0/2

2/2

-1/1

-1/1

1/1



Question 40 Le compilateur PTS est écrit en PTS.

☐ vrai

☒ faux

Question 41 Le compilateur PTS produit du code pour une machine virtuelle.

☐ faux

☒ vrai

Question 42 Le code produit par le compilateur PTS doit être traduit en code machine avec GCC pour être exécuté.

☒ faux

☒ vrai

Question 43 DOT est un traducteur.

☒ vrai

☒ faux

Question 44 DOT accepte en entrée un programme Java.

☐ vrai

☒ faux

Question 45 DOT est utilisé dans le compilateur PTS pour afficher les arbres syntaxiques abstraits produits au cours de la compilation.

☒ vrai

☐ faux

Question 46 La machine MAP est un interpréteur.

☐ faux

☒ vrai