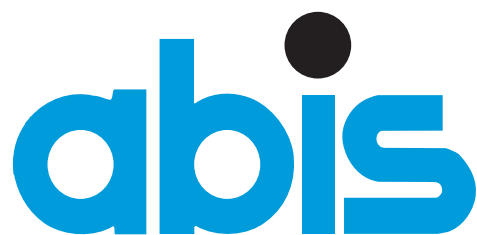


# Auto-évaluation Programmation en PL/1

Document: f0830test.fm

21 août 2023

ABIS Training & Consulting  
Diestsevest 32 / 4b  
B-3000 Leuven  
Belgium

The logo for ABIS Training & Consulting. The word "abis" is written in a bold, blue, lowercase sans-serif font. A solid black circle is positioned above the letter 'i'. Below the text is a thick, horizontal grey bar.

TRAINING & CONSULTING

# INTRODUCTION AUTO-ÉVALUATION PROGRAMMATION EN PL/1

Ce test vous permettra d'estimer vos connaissances préalables pour le cours [La programmation en PL/1: cours de base](#). Il est basé sur les sujets traités dans ce cours.

Ce test comporte 20 questions à choix multiples. Pour la plupart des questions, plusieurs réponses correctes sont possibles. Cochez les solutions qui vous semblent bonnes, et comparez vos réponses avec les solutions.

Il n'y a pas de question piège (du moins pas consciemment), mais lisez les questions et les propositions bien attentivement. Comptez environ 15 minutes pour le remplir.

Les bonnes réponses se trouvent à la fin de ce présent document.

# QUESTIONS AUTO-ÉVALUATION

## PROGRAMMATION EN PL/1

1. Quelles sont, parmi celles-ci, les propositions qui sont VRAIES? (2 réponses)

- [a] Un programme PL/1 doit toujours commencer par une instruction PROCEDURE et se terminer par une instruction END.
- [b] Les instructions PL/1 peuvent être codées entre la position 1 et 72.
- [c] Toute variable que l'on désire utiliser dans un programme PL/1 doit être déclarée.
- [d] Un programme PL/1 peut être constitué de plusieurs procédures.
- [e] Dans un programme PL/1, une ligne de commentaire est commencée par //.

2. Un programmeur PL/1 veut exécuter une instruction 5 fois de suite. Comment doit-il coder cela?

(a)

```
REPEAT 5 TIMES;  
    instruction;  
END;
```

(b)

```
DO J=1 TO 5;  
    instruction;  
END;
```

(c)

```
COUNTER = 1;  
DO WHILE COUNTER < 5;  
    COUNTER = COUNTER + 1;  
    instruction;  
END;
```

(d)

```
REPEAT VARYING COUNTER FROM 1 BY 1 UNTIL COUNTER = 5;  
    instruction;  
END;
```

3. Lesquels, parmi ces noms de variables pour data items, sont syntaxiquement corrects? (2 réponses)
- [a] READ
  - [b] RECORD-IN
  - [c] @\_RECORD
  - [d] IN/OUTRECORD
  - [e] 1\_TELEPHONE
4. Où est-on obligé de coder un ; (**point-virgule**) dans un programme PL/1? (3 réponses)
- [a] après chaque déclaration de variable
  - [b] à la fin de chaque instruction
  - [c] pour clôturer une instruction DO
  - [d] à la fin d'un programme
  - [e] pour clôturer une instruction IF
5. Supposons que l'on désire faire appel, dans un programme PL/1, à une procédure externe qui attend 1 paramètre et qui renvoie 1 valeur. De quelles instructions a-t-on besoin dans le programme principal? [3 réponses]
- [a] DCL MYSUB ENTRY (CHAR(5)) EXTERNAL;
  - [b] DCL MYSUB ENTRY (CHAR(5)) RETURNS (CHAR(5)) EXTERNAL;
  - [c] DCL MYPARAMETER, MYRESULT CHAR(5);
  - [d] CALL MYSUB (MYPARAMETER);
  - [e] MYRESULT = MYSUB;
  - [f] MYRESULT = MYSUB(MYPARAMETER);

6. Considérons le programme PL/1 suivant. Supposons que A, B, C, ... sont des instructions PL/1. Dans quel ordre les instructions seront-elles exécutées?

```
MYPROG : PROCEDURE OPTIONS (MAIN);
  CALL INIT;
  Z;
  CALL TERM;
  Z;
END MYPROG;

INIT : PROCEDURE;
  A;
  B;
  CALL READ-X;

LOOP : PROCEDURE;
  E;
  F;

TERM : PROCEDURE;
  G;
  H;

READ-X : PROCEDURE.
  X;
```

- (a) A B Z E F Z X
- (b) A B X Z E F G H Z X
- (c) A B X Z E F Z G H Z
- (d) A B X Z G H Z
7. Considérons l'opération suivante. En tenant compte des règles de priorité, comment cette opération va-t-elle être exécutée?

$A ** 2 * B / 2 + c - 5$

- (a)  $((A ** 2) * B) / 2 + c - 5$
- (b)  $((A ** 2) * (B / 2)) + c - 5$
- (c)  $(A ** 2) * (B / (2 + c)) - 5$
- (d)  $((A ** (2 * B)) / 2) + c - 5$

8. Un programmeur PL/1 désire qu'un nombre soit affiché dans le même format que l'exemple ci-dessous. Laquelle de ces définitions de variable permet-elle cela?

```
+123.45
```

(il est supposé que le signe décimal est le '.')

- (a) PIC '(5)9'
  - (b) PIC 'S(3)9V.99'
  - (c) PIC 'S999.99'
  - (d) PIC '999V99'
9. Considérons les déclarations de variables suivantes. Quel est le contenu de RESULT à la fin des instructions suivantes?

```
DCL 1 VAR1      CHAR(5) INIT ('PL/1');  
DCL 1 VAR2      CHAR(10);  
DCL 1 VAR3      CHAR (2);  
DCL 1 RESULT    CHAR (5);
```

```
RESULT = 'COBOL';  
VAR2 = VAR1;  
VAR3 = VAR2;  
RESULT = VAR3;
```

- (a) PLbbb (bbb = 3 blancs)
  - (b) COBOL
  - (c) COBPL
  - (d) PLBOL
10. Considérons, les variables PL/1 suivantes. Laquelle de ces conditions n'est-elle PAS correcte?

```
DCL  INFILE  FILE RECORD INPUT;  
DCL  EOF     BIT(1)  INIT ('0'B);
```

```
ON ENDFILE (INFILE)  
  EOF = '1'B;
```

- (a) DO WHILE (NOT EOF);
- (b) DO WHILE (^EOF);
- (c) DO WHILE (EOF = '0'B);
- (d) DO WHILE (EOF ^= '1'B);

11. Quelles définitions peut-on utiliser pour la description d'un enregistrement d'un fichier séquentiel qui doit être directement utilisé en INPUT? (2 réponses)

- [a] CHAR (9)
- [b] BIN FIXED (15)
- [c] PIC '(4)Z'
- [d] PIC '9999V99'
- [e] PIC '9999V.99'

12. Laquelle de ces instructions SELECT en PL/1 est-elle correcte?

(a)

```
SELECT (A);  
    WHEN (A=0)      PUT LIST('A = 0');  
    WHEN (A=5)      PUT LIST ('A = 5');  
    OTHERWISE      PUT LIST ('A IS NOT 0 NOR 5');  
END;
```

(b)

```
SELECT (A);  
    WHEN (0)        PUT LIST('A = 0');  
    WHEN (5)        PUT LIST ('A = 5');  
    OTHERWISE      PUT LIST ('A IS NOT 0 NOR 5');  
END;
```

(c)

```
SELECT;  
    WHEN (A=0)      PUT LIST('A = 0');  
    WHEN (5)        PUT LIST ('A = 5');  
    OTHERWISE      PUT LIST ('A IS NOT 0 NOR 5');  
END;
```

13. Quelles déclarations de variables sont-elles correctes? (2 réponses)

- [a] DCL VAR\_4, CHAR;
- [b] DCL VAR\_1 CHAR(32800);
- [c] DCL (VAR\_7, VAR\_8) CHAR(4);
- [d] DCL CHAR(4);

14. Observons les instructions PL/1 suivantes. Quand PAR\_3 est-il exécuté?

```
IF A=B THEN
  PAR_1;
ELSE
  PAR_2;
  PAR_3;
```

- (a) jamais
- (b) si A et B sont égaux
- (c) si A et B sont différents
- (d) toujours

15. Laquelle parmi ces affirmations à propos des arrays en PL/1 est-elle VRAIE?

- (a) Il est possible de définir des arrays multidimensionnelles. Le nombre de dimensions est limité à 15.
- (b) Les éléments d'une array peuvent seulement être du type numérique.
- (c) On peut faire référence à une élément d'une array au moyen d'un indice. Le premier élément d'une array a comme indice 0.
- (d) Les arrays sont utilisées pour lire des tables DB2

16. Pour chaque fichier que l'on désire lire dans un programme PL/1, il faut faire un certain nombre de déclarations et coder un certain nombre d'instructions. Dans quelle ordre, ces instructions doivent-elles être exécutées?

```
(1) DCL infile FILE RECORD INPUT
(2) READ FILE (infile)
(3) OPEN FILE (infile)
(4) CLOSE FILE (infile)
```

- (a) 3, 1, 2, 4
- (b) 1, 3, 2, 4
- (c) 1, 2, 3, 4
- (d) 3, 2, 1, 4



17. Quel est le contenu de RESULT après l'exécution des instructions PL/1 suivantes. La valeur de départ de RESULT est 0.

```
RESULT = 0;  
DO C=1 BY 1 UNTIL (C=10);  
    RESULT = RESULT + 1;  
END;
```

Réponse: .....

18. Quelle est la différence en PL/1 entre un bloc BEGIN et un bloc PROCEDURE?

- (a) un bloc PROCEDURE est activé à chaque fois qu'il est appelé par une instruction CALL; un bloc BEGIN est toujours exécuté.
- (b) une variable qui est déclarée dans un bloc BEGIN est toujours utilisable dans tout le programme entier; en ce qui concerne un bloc PROCEDURE, la variable est uniquement accessible dans la PROCEDURE elle-même.
- (c) un bloc BEGIN peut être quitté au moyen d'une instruction END; on peut quitter un PROCEDURE au moyen d'une instruction RETURN

19. Laquelle de ces déclarations d'array est-elle correcte?

- (a) DCL ARRAY\_1 DEC FIXED (3)(5,2);
- (b) DCL ARRAY\_1 (5,2) DEC FIXED (3);
- (c) DCL ARRAY\_1 (5) DEC FIXED (3), (2) DEC FIXED (3);
- (d) DCL (5,2) ARRAY\_1 DEC FIXED (3);

20. Supposons que l'on lise dans un programme PL/1 une variable directement via un stream-input. Considérons les déclarations suivantes et les instructions GET. A quoi doivent ressembler les variables de cet input-stream?

```
DCL  NUM_1      DEC FIXED (5,2);
DCL  NUM_2      DEC FIXED (5,2);
DCL  AN_1       CHAR(6);
```

```
GET EDIT (NUM_1,NUM_2,AN_1) (COL(1),F(5,2),COL(10),F(5,2),COL(20),A(6));
```

```
      1---5----0----5----0----5----0----5----0----5----0----5
(1) 1254      1200      ABCDE
(2) 12.54     12.00     ABCDE
(3) 1254      1200     'ABCDE'
```

- (a) 1
- (b) 2
- (c) 3
- (d) Les trois sont en ordre

# EVALUATION.

Ici sont les réponses correctes:

1. a d
2. b
3. a c
4. b d e
5. b c f
6. d
7. a
8. b
9. a
10. a
11. a d
12. b
13. a c
14. d
15. a
16. b
17. 10
18. a
19. b
20. a

Comptez 1 point par bonne réponse. Pour les questions avec plusieurs bonnes réponses, comptez 1 point seulement si vous avez coché toutes les bonnes alternatives.

Si votre score atteint 80% ou plus, il ne vous est pas nécessaire de suivre ce cours.

Si votre score est entre 50% et 80%, le cours [La programmation en PL/1: cours de base](#) vous permettra de compléter vos connaissances.

Si votre score est inférieur à 50%, il vous est vivement conseillé de suivre le cours [La programmation en PL/1: cours de base](#).