

NOM :
Prénom :

Salle :
Filière :

N° C.I.N.
N° d'inscription

Examen

Session de contrôle

Année universitaire : 2014-2015

Matière : Programmation I, Filière : MPI
Enseignants : W. Drira
Nombre de pages : 4

Date: Juin 2015

Durée: 1h 30

Documents: NON autorisés

Attention, les réponses qui seront prises en compte sont celles reportées dans le tableau de réponses page 4.

1. Le programme chargé de traduire votre code d'un langage de haut niveau vers le langage binaire est appelé :
A. Le compilateur
B. Le traducteur
C. Le linéur
D. Le programmeur
2. Quelle est la fonction qui permet d'afficher du texte à l'écran en mode console ?
A. Aff("Bonjour ");
B. print("Bonjour ");
C. printf("Bonjour ");
D. fprintf("Bonjour ");
E. write("Bonjour ");
3. Quel symbole permet d'effectuer un retour du curseur à gauche de l'écran ?
A. \a
B. \n
C. \t
D. \r
4. Quel type de donnée permet de stocker 3.142 ?
A. int
B. double
C. long
D. char
5. Combien de fois passe-t-on dans la boucle suivante ?

```
int a = 10;  
do{a++;printf("bonjour\n");}while (a<10);
```


A. 0
B. 1
C. 9
D. 10
6. Lorsqu'un programme utilise *strlen* il faut qu'il contienne l'instruction préprocesseur
A. #include <string.h>
B. #include <stdio.h>
C. #include <stdlib.h>
D. #include <ctype.h>
7. Lorsqu'un programme utilise *toupper* il faut qu'il contienne l'instruction préprocesseur:
A. #include <string.h>
B. #include <stdio.h>
C. #include <stdlib.h>
D. #include <ctype.h>

8. Quelle étape de la compilation vient d'échouer lorsqu'on a un message comme celui-ci :
Undefined symbols : "_printf"= ou =référence indéfinie vers «printf»
A. l'analyse sémantique
B. l'édition de liens
C. l'analyse harmonique
D. l'analyse des entrées clavier
9. Laquelle des analyses suivantes ne fait pas partie des étapes de la compilation :
A. analyse harmonique
B. analyse syntaxique
C. analyse sémantique
D. analyse lexicale
10. On considère les déclarations suivantes :

```
#include <stdio.h>  
int fonction(int z ) {  
    return z*z ;}  
int main() {  
    int x;  
    fonction(x) ;}
```


A. "x" est une variable locale
B. "x" est une variable globale
C. "x" est un paramètre réel
D. "x" est un paramètre formel
E. "x" est une constante
11. Qu'affichent les instructions suivantes :

```
#include <stdio.h>  
int N=11;  
int main() {  
    int N=2;  
    if(N>5)  
        printf(" %d",N);  
    printf(" %d",N);}
```


A. 2
B. 11
C. 11 2
D. 2 11
E. Autre
12. Le code suivant :

```
if(x) { bloc_0 ; } bloc_1 ;
```


A. exécute le bloc_1 si x vaut 0
B. exécute le bloc_0 si x vaut -2
C. provoque une erreur si x est de type float
D. est syntaxiquement faut : il manque le « else »

NOM :

Salle :

N° C.I.N.

Prénom :

Filière :

N° d'inscription

13. Le code suivant :

```
if(N)
printf(" Bon"); printf("jour ");
else
printf(" Bonsoir ");
A. Affiche « bonjour » si N vaut -5
B. Affiche « bon » si N vaut -5
C. Affiche « Bonsoir » si N vaut 0
D. Provoque une erreur
```

14. Les lignes :

```
int i, n = 7;
for(i = 0; i < 4; i++)
{ n = n + 2*i;
printf("%d ", n); }
A. Affiche 7 9 13 19
B. Comportent une erreur qui sera détectée
au cours de l'analyse syntaxique
C. Comportent une erreur qui sera détectée
au cours de l'édition de lien
D. Comportent une erreur qui ne sera pas
détectée
```

15. Quelle est le résultat du code suivant :

```
char s[]="abcd";
sprintf(s,"%s %d %c %d
%s",s,11,':',55,s);
printf("%s",s);
A. abcd
B. abcd abcd 11:55
C. abcd abcd 11:55 abcd
D. abcd 11:55 abcd 11:55
E. Erreur
```

16. Lesquelles de ces définitions de fonctions sont correctes si elles sont sensées calculer le factoriel de N ?

- A. float Factoriel (float N)
- B. int Factoriel (float N)
- C. int Factoriel (N)
- D. int Factoriel (int N)

17. Le code suivant :

```
if(a) code_1; code_2;
A. exécute code_1 si a vaut -2
B. exécute code_1 et code_2 si a vaut -1
C. exécute code_2 si a vaut 0
D. exécute code_1 et code_2 si a vaut 0
E. provoque une erreur
```

18. Quelle est la sortie à l'écran du programme suivant ?

```
#include <stdio.h>
char c;
void Affect_char() {char c = 69 ; }
int main() {
c = 84 ;
Affect_char();
```

```
printf("%c",c) ;
getch();
return 1;}
```

- A. T
- B. E
- C. 84
- D. 69
- E. Erreur

19. Où sont déclarées les variables locales ?

- A. En début du programme principal.
- B. Au début de chaque fonction où elles interviennent.
- C. Elles sont passées en paramètres dans l'énoncé de la fonction

20. On considère l'appel suivant :

```
incrementation(A);
```

Pour qu'il y ait incrémentation de la valeur du variable A dans le programme principal, on utilise :

- A. Un passage par valeur.
- B. Un passage par adresse.

21. Où sont déclarées les fonctions ?

- A. Après la fonction principale « main », si les prototypes sont définis avant
- B. Au cours du programme principal.
- C. Avant le programme principal et la règle "une fonction doit être déclarée avant usage" doit être respectée.

22. Considérons l'entête de la fonction suivante :

```
int Ma_focntion(float *a,float *b,int c,char
d,char e);
```

Considérons les déclarations suivantes :

```
char u,v; float x, y ; int z ;
```

Quels sont les appels de fonction justes ?

- A. Ma_focntion(x,y,z,u,v);
- B. Ma_focntion(&x,&y,z,u,v);
- C. Ma_focntion(&x,&y,'5',u,v);
- D. Ma_focntion(&x,&y,z,'u',v);
- E. int Ma_focntion(&x,&y,z,u,v);

23. On considère les déclarations suivantes :

```
#include <stdio.h>
int triple(int z) {return z*3 ;}
int main() {
int a;
triple(a); }
```

- A. "z" est une variable locale
- B. "z" est une variable globale
- C. "z" est un paramètre formel
- D. "z" est un paramètre réel

24. Qu'affiche le programme suivant ?

```
void fct1 (int x) { x = x + 10; printf
("%d ", x);}
int fct2 (int y) { return y + 20;}
```



```
int main (void) {
    int x = 1;
    fct1 (x);
    printf ("%d ", x);
    x = fct2 (x);
    printf ("%d\n", x);
    getch();
    return 0;}
```

- A. 11 11 31
- B. 11 1 31
- C. 11 11 21
- D. 11 1 21

25. Qu'affiche le programme suivant ?

```
void fct(int a[]) {a[1] = 10;}
int main(void) {
    int T[]={1,2,3};
    fct(T);
    printf("%d", T[1] );
    getch();
    return 0;}
```

- A. 1
- B. 2
- C. 10
- D. Erreur

26. Que fait la fonction strncpy(S1, S2, 5)?

- A. Ajouter les 5 premiers octets de S2 à S1
- B. Ajouter les 5 premiers octets de S1 à S2
- C. Copier les 5 premiers octets de S1 dans S2
- D. Copier les 5 derniers octets de S2 dans S1
- E. Copier les 5 premiers octets de S2 dans S1

27. Choisir la structure de boucle adéquate : Si le nombre d'itérations n'est a priori pas connu, on utilise :

- A. while...
- B. do... while
- C. for...

28. Qu'affiche le programme suivant ?

```
int A=3,B=5,X=8,Y=12,r ;
r=Y+B ; X=A-r ; B+=A+X ; A++ ;
printf("%d %d %d %d %d",A,B,X,Y,r) ;
```

- A. 4 -6 -14 17 17
- B. 4 -6 -14 12 17
- C. 4 17 -8 29 3
- D. 4 17 1 29 3
- E. Erreur

29. On considère le programme suivant :

```
int Tab[3],i;
for(i=0;i<3;i++)
    Tab[i]=i++;
for(i=0;i<3;i++)
    printf("%d ",Tab[i]) ;
```

- A. Il affiche 0 1 2
- B. Il affiche 0 2 4
- C. Il affiche 1 2 3
- D. Il affiche 0, une valeur imprévisible et 3
- E. Il affiche 0, une valeur imprévisible et 2

30. Qu'affiche le programme suivant :

```
int x= 7;
void fct(int x){ printf(" %d ",x*=2);}
int main(){
    int x=6;
    fct(x);
    printf(" %d ",x) ;
    getch();
    return 1;}
```

- A. 12 6
- B. 12 12
- C. 14 6
- D. 14 7
- E. 14 14

31. Après la définition :

```
int a, M[4],N[3] ;
```

parmi les expressions suivantes, lesquelles sont syntaxiquement incorrectes (c à d qui provoque une erreur de compilation)

- A. a = M ;
- B. a = N-M ;
- C. M=N ;
- D. a = M[3] ;
- E. a = 1[M] ;

32. Après la définition :

```
typedef float reel;
typedef int entier;
reel r ;
entier x, y;
```

Lesquelles de ces déclarations sont justes ?

- A. reel M[10][y] ;
- B. reel M[x][y] ;
- C. char M[1][r] ;
- D. entier M[]={do, re, me, fa} ;
- E. entier M[x][y] ;

33. Qu'affiche le programme suivant :

```
int i= 13;
while(i>0){i=i-3;printf("%d ",i);}
```

- A. 7 4 1
- B. 7 4 1 -2
- C. 10 7 4 1
- D. 10 7 4 1 -2
- E. 13 10 7 4 1

34. On considère le programme suivant ?

```
int a = 3, b = 9, c = 6, n;
n=(a>b)?((a>c)?'a':'c'):(b>c)?'b':'c');
```

- A. Il affecte la valeur 6 à la variable n
- B. Il affecte la valeur 9 à la variable n
- C. Il affecte le code ASCII du caractère c à la variable n
- D. Il affecte le code ASCII du caractère b à la variable n

35. Pour que la variable a prend la valeur de b si a est impaire, sinon elle sera divisée par 3, on peut écrire :

- A. a=((a%2)) ? a/3 : b ;

NOM :
Prénom :

Salle :
Filière :

N° C.I.N.
N° d'inscription

- B. $a = ((a \% 2)) ? b : a / 3$;
C. $a /= ((a \% 2) == 0) ? 3 : b$;
D. $\text{if } (a \% 2 != 0) a = b ; \text{ else } a = a / 3 ;$

Reportez vos réponses dans le tableau ci-dessous

36. Parmi les séquences suivantes, lesquelles sont équivalentes à la boucle suivante:

- for (i=a; i<b; i++) { f(i); }
A. i=a; if(a<b){ do { f(i); i++; } while(i<b); }
B. i=a; while(i<b){ f(i++); }
C. i=a; do { f(i++); } while(i<b);
D. i=a; while(i<b){ f(i); i++; }
E. i=a; while(i<b){ f(i); } i++;

37. Qu'affiche le programme suivant :

```
int a= 25, b=10, c, d;  
float x, y, z;  
c=a/b; d=b/a; x=a/b; y=b/a; z=a/x;  
printf("%d %d %d %d %2.1f %2.1f  
%2.1f", a,b,c,d,x,y,z);
```

- A. 25 10 2 0 2.5 0.4 10
B. 25 10 2 0 2.0 0.0 12.5
C. 25 10 2 0 2.0 0.0 13.0
D. 25 10 3 0 2.0 0.0 12.5

38. Lesquelles de ces déclarations sont justes ?

- A. `typedef enum M[]={do,ré,mi};`
B. `typedef enum M {do,ré,mi};`
C. `typedef enum M{"do","ré","mi"};`
D. `entier M[]={do,ré,mi} ;`
E. Aucune

39. Soit A un tableau de 8 entiers, lesquelles de ces instructions permettant de saisir au clavier le 4^{ème} entier.

- A. `scanf("%d", &T1 + 3);`
B. `scanf("%d", T1 + 3);`
C. `scanf("%d", &T1[3]);`
D. `scanf("%d", &T1[4]);`
E. `scanf("%d", &T1[0] + 3);`

40. Quelle est l'instruction (une seule instruction) qui permet de copier les 5 premiers caractères de la chaîne de caractère A1 dès la position du 2^{ème} caractère de la chaîne de caractère A2. Le résultat sera affecté à A.

1	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
2	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
3	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
4	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
5	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
6	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
7	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
8	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
9	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
10	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
11	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
12	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
13	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
14	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
15	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
16	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
17	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
18	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
19	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
20	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
21	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
22	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
23	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
24	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
25	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
26	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
27	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
28	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
29	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
30	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
31	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
32	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
33	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
34	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
35	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
36	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
37	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
38	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
39	A. <input type="checkbox"/> , B. <input type="checkbox"/> , C. <input type="checkbox"/> , D. <input type="checkbox"/> , E. <input type="checkbox"/>
40