

QCM de langage C (LC4)  
Mercredi 7 mars 14, et 7  
Durée : 1h, Documents, ordinateurs interdits.  
Téléphones éteints

Pour chacune des 40 questions, il y a au moins une bonne réponse et pour 4, il y en a plusieurs.  
Il y a 48 bonnes réponses au total pour 293 choix possibles.  
Notation : un choix correct 1pt, un choix incorrect -0,5pt, absence de réponse 0pt.  
La note (maximum 48) sera ensuite ramenée à une note sur 20

Question 1. Si on utilise la directive

```
#include "truc.h"
```

le fichier sera recherché

- ☐ 1) seulement dans les répertoires spécifiés par l'utilisateur
- ☐ 2) dans les répertoires système, puis dans les répertoires spécifiés par l'utilisateur
- ☐ 3) d'abord dans les répertoires spécifiés par l'utilisateur, puis dans les répertoires système
- ☐ 4) seulement dans les répertoires système

Question 2. Après la macro-définition

```
#define f(x,y) x*y
```

et l'exécution de `z = f(3-6, 4+2);`

- |                                 |  |                                 |   |
|---------------------------------|--|---------------------------------|---|
| <input type="checkbox"/> 1) -19 | <input type="checkbox"/> 2) erreur à l'exécution | <input type="checkbox"/> 3) -18 | <input type="checkbox"/> 4) erreur à la compilation |
| <input type="checkbox"/> 5) -24 | <input type="checkbox"/> 6) -33                  | <input type="checkbox"/> 7) -10 | <input type="checkbox"/> 8) -23                     |

Question 3. Après la séquence de code suivante :

```
#define T int *
```

```
T x, y;
```

- |  |   |
|--|---|
| <input type="checkbox"/> 1) x est de type int * et y est de type int | <input type="checkbox"/> 2) erreur à l'exécution                                  |
| <input type="checkbox"/> 3) x est de type int et y est de type int * | <input type="checkbox"/> 4) les variables x et y sont toutes les deux de type int |
| <input type="checkbox"/> 5) erreur à la compilation                  | <input type="checkbox"/> 6) x et y sont toutes les deux de type int *             |

Question 4. Pour définir une constante numérique GOLD pour le nombre d'or avec 6 décimales au moyen d'une macro, on peut écrire :

- |   |   |  |
|---|---|--|
| <input type="checkbox"/> 1) <code>#define GOLD=1.618034;</code> | <input type="checkbox"/> 2) <code>#define GOLD&lt;-1.618034;</code> | <input type="checkbox"/> 3) <code>#define GOLD 1.618034;</code>  |
| <input type="checkbox"/> 4) <code>#define GOLD 1.618034</code>  | <input type="checkbox"/> 5) <code>#define GOLD "1.618034";</code>   | <input type="checkbox"/> 6) <code>#define GOLD "1.618034"</code> |
| <input type="checkbox"/> 7) <code>#define GOLD=1.618034</code>  | <input type="checkbox"/> 8) <code>#define GOLD&lt;-1.618034</code>  |  |

**Question 5.** La séquence complète de construction d'un binaire exécutable est constituée des phases suivantes dans cet ordre :

- ☐ 1 préprocessing, compilation, assemblage, édition de liens, postprocessing
- ☐ 2 compilation, assemblage, édition de liens, postprocessing
- ☐ 3 préprocessing, édition de liens, compilation, assemblage
- ☐ 4 préprocessing, assemblage, compilation, édition de liens
- ☐ 5 préprocessing, compilation, assemblage, édition de liens
- ☐ 6 compilation, préprocessing, assemblage, édition de liens
- ☐ 7 préprocessing, compilation, éditions de liens, assemblage

**Question 6.** Quelle(s) qualification(s) limite(nt) dans une définition la portée d'une variable globale au fichier dans lequel cette définition se trouve ?

- |                                       |                                     |                                     |
|---------------------------------------|-------------------------------------|-------------------------------------|
| <input type="checkbox"/> 1 extern     | <input type="checkbox"/> 2 register | <input type="checkbox"/> 3 auto     |
| <input type="checkbox"/> 4 restricted | <input type="checkbox"/> 5 limited  | <input type="checkbox"/> 6 volatile |
| <input type="checkbox"/> 7 local      | <input type="checkbox"/> 8 static   | <input type="checkbox"/> 9 const    |

**Question 7.** Que provoque la séquence suivante

```
int i = 4, j;  
j = ((++i)--)+;
```

- |  |   |   |
|--|---|---|
| <input type="checkbox"/> 1 erreur à la compilation | <input type="checkbox"/> 2 i vaut 4 et j vaut 4 | <input type="checkbox"/> 3 i vaut 5 et j vaut 4 |
| <input type="checkbox"/> 4 erreur à l'exécution    | <input type="checkbox"/> 5 i vaut 5 et j vaut 5 | <input type="checkbox"/> 6 i vaut 4 et j vaut 5 |

**Question 8.** Que peut-on dire du programme suivant ?

```
#include <stdio.h>  
int main() {  
    int m = 0x328, n = -m;  
    printf("%x %x %x\n", m&n, m|n, m^n);  
    return 0;  
}
```

- |   |   |
|---|---|
| <input type="checkbox"/> 1 il a che 0 fffff328 fffffde8 | <input type="checkbox"/> 2 il a che 328 1ffff328 fffffde8 |
| <input type="checkbox"/> 3 erreur à la compilation      | <input type="checkbox"/> 4 il a che 8 1ffff328 fffffde8   |
| <input type="checkbox"/> 5 erreur à l'exécution         | <input type="checkbox"/> 6 il a che 328 1ffff328 10000000 |
| <input type="checkbox"/> 7 il a che 8 fffffff8 fffffff0 | <input type="checkbox"/> 8 il a che 328 fffff328 fffffde8 |

**Question 9.** Que produit la séquence :

```
int n = 0x8A6B87;  
printf("%x\n", n >> 3);
```

- |  |  |   |
|--|--|---|
| <input type="checkbox"/> 1 a chage de 8A6B87 | <input type="checkbox"/> 2 a chage de 114170 | <input type="checkbox"/> 3 a chage de B87 |
| <input type="checkbox"/> 4 a chage de 114d70 | <input type="checkbox"/> 5 a chage de 8A6B80 | <input type="checkbox"/> 6 a chage de 86A |

**Question 10.** Pour ajouter à *a* la valeur *b* si *a* est un multiple de 3 et multiplier *a* par 2 sinon, on peut écrire

- |   |   |
|---|---|
| <input type="checkbox"/> 1 <code>a = ((a % 3) == 0) ? 2 * a : a + b;</code> | <input type="checkbox"/> 2 <code>if(a % 3) a += b; else a *= 2;</code>        |
| <input type="checkbox"/> 3 <code>a = (a % 3) ? 2 * a : a + b;</code>        | <input type="checkbox"/> 4 <code>a = (! (a % 3) == 0) ? 2 * a : a + b;</code> |
| <input type="checkbox"/> 5 <code>a = (!a % 3) ? 2 * a : a + b;</code>       | <input type="checkbox"/> 6 <code>a = (a % 3 == 1) ? 2 * a : a + b;</code>     |

---

**Question 11.** Que produit le programme suivant ?

```
#include <stdio.h>
int main() {
    int x = 8, n;
    n = (x++, --x, ++x, x--);
    printf("%d %d\n", n, x);
    return 0;
}
```

- |                                      |                                      |  |                                      |
|--------------------------------------|--------------------------------------|--|--------------------------------------|
| <input type="checkbox"/> 1 a che 7 8 | <input type="checkbox"/> 2 a che 9 7 | <input type="checkbox"/> 3 a che 8 7               | <input type="checkbox"/> 4 a che 9 8 |
| <input type="checkbox"/> 5 a che 7 9 | <input type="checkbox"/> 6 a che 9 9 | <input type="checkbox"/> 7 erreur à la compilation | <input type="checkbox"/> 8 a che 8 8 |

---

**Question 12.** Que peut-on dire de la séquence suivante ?

```
int a = 9, b = 6, c = 8;
int n = (a > b) ? ((b > c) ? 'b' : 'c') : ((a > c) ? 'c' : 'a');
```

- |  |  |
|--|--|
| <input type="checkbox"/> 1 a ecte la valeur 9 à la variable <i>n</i>                         | <input type="checkbox"/> 2 provoque une erreur de compilation                                |
| <input type="checkbox"/> 3 a ecte la valeur 6 à la variable <i>n</i>                         | <input type="checkbox"/> 4 a ecte la valeur 8 à la variable <i>n</i>                         |
| <input type="checkbox"/> 5 a ecte le code ASCII du caractère <i>c</i> à la variable <i>n</i> | <input type="checkbox"/> 6 a ecte le code ASCII du caractère <i>b</i> à la variable <i>n</i> |
| <input type="checkbox"/> 7 a ecte le code ASCII du caractère <i>a</i> à la variable <i>n</i> |  |

---

**Question 13.** Que produit le programme suivant ?

```
#include <stdio.h>
int main(){
    int a = 5, b = 3, c;
    c = --a + b-- ;
    printf("%d %d %d\n", a, b, c);
    return 0;
}
```

- |  |  |  |
|--|--|--|
| <input type="checkbox"/> 1 a che 5 2 6                     | <input type="checkbox"/> 2 a che 5 2 8 | <input type="checkbox"/> 3 une erreur à la compilation |
| <input type="checkbox"/> 4 a che des valeurs imprévisibles | <input type="checkbox"/> 5 a che 4 2 6 | <input type="checkbox"/> 6 a che 4 2 7                 |
| <input type="checkbox"/> 7 a che 5 2 7                     | <input type="checkbox"/> 8 a che 5 3 6 | <input type="checkbox"/> 9 a che 4 2 8                 |

**Question 14.** Que produit le programme suivant ?

```
#include <stdio.h>
int t[] = {2, 1, 4, 5, 0};
int main(){
    int i;
    for(i = 0; i < 5; i++){
        switch (t[i]){
            case 1 : printf("a");
            case 2 : printf("b");
            case 3 : printf("c"); break;
            case 4 : printf("d"); break;
            case 5 : printf("e");
            default : printf("f");
        }
    }
    return 0;
}
```

- |  |   |  |
|--|---|--|
| <input type="checkbox"/> 1 a che baddef            | <input type="checkbox"/> 2 erreur à l'exécution | <input type="checkbox"/> 3 a che bcdefabcdefdefeff |
| <input type="checkbox"/> 4 erreur à la compilation | <input type="checkbox"/> 5 a che badef          | <input type="checkbox"/> 6 a che bcabcdeff         |
| <input type="checkbox"/> 7 a che bcadeff           | <input type="checkbox"/> 8 a che bcdadef        |  |

---

**Question 15.** Que produit l'instruction suivante ?

```
printf("%d\n", 0 ? 7 : 6);
```

- |                                    |   |  |
|------------------------------------|---|--|
| <input type="checkbox"/> 1 a che 6 | <input type="checkbox"/> 2 a chage imprévisible | <input type="checkbox"/> 3 erreur à la compilation |
| <input type="checkbox"/> 4 a che 0 | <input type="checkbox"/> 5 a che 7              |  |

---

**Question 16.** Que produit la séquence de code suivante ?

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

- |   |  |   |
|---|--|---|
| <input type="checkbox"/> 1 a che 7 4 1          | <input type="checkbox"/> 2 a che 13 10 7 4 1 | <input type="checkbox"/> 3 a che 10 6     |
| <input type="checkbox"/> 4 a che 10 7 4 1 -2    | <input type="checkbox"/> 5 a che 10 7 4 1    | <input type="checkbox"/> 6 a che 7 4 1 -2 |
| <input type="checkbox"/> 7 a che 13 10 7 4 1 -2 |  |   |

---

**Question 17.** Que produit le programme suivant ?

```
#include<stdio.h>
int main() {
    int _ = 5;
    int __ = 10;
    int ___;
    ___ = _ + __;
    printf("%dae!5;
```

**Question 18.** Que produit le programme suivant :

```
#include <stdio.h>
int main() {
    int i,j;
    for(i = 0; ; i++) {
        for(j = 0; ; j++)
            if(j > 100) break;
        printf("%d %d ", i, j);
        if(i == 2) break;
    }
    return 0;
}
```

- ☐ 1 a che 0 101 1 101 2 101      ☐ 2 a che 0 101 1 101      ☐ 3 a che 0 100 1 100 2 100  
☐ 4 boucle infinie      ☐ 5 erreur à la compilation      ☐ 6 a che 100 101  
☐ 7 a che 0 101

---

**Question 19.** Que produit le programme suivant ?

```
#include <stdio.h>
main(){
    int i, j;
    for(i = 1; i <= 3; i++){
        static int p = 1;
        static int s = 0;
        p += i;
        for(j = 0; j < 2; j++)
            s += p;
        printf("%d %d ", p, s);
    }
    return 0;
}
```

- ☐ 1 a che 2 4 4 8 7 14      ☐ 2 a che 2 4 3 6 4 8      ☐ 3 a che 2 4 4 12 7 26  
☐ 4 erreur à l'exécution      ☐ 5 a che 2 4 3 10 4 18      ☐ 6 erreur à la compilation
-

Question 20. Quel message le programme affiche-t-il à l'exécution?

```
#include <stdio.h>
int a = 10;
int main(){
    int i;
    for(i = 0; i < 2; i++){
        a++;
        { static int a = 2;
          a += i;
          printf("%d ", a);
        }
        a++;
        printf("%d ", a);
        { int a = 1;
          a++;
          printf("%d ", a);
        }
    }
    printf("%d ", a);
    return 0;
}
```

☐ 1 a che 2 12 2 3 12 2 10

☐ 2 a che 11 12 13 15 16 17 17

☐ 3 erreur à l'exécution

☐ 4 erreur à la compilation

☐ 5 a che 2 12 2 3 14 2 10

☐ 6 ac the 2 12 2 3 14 2 14

☐ 7

**Question 23.** Si `f` est une fonction ayant comme paramètres un entier et un pointeur sur entier, et ne renvoyant pas de valeur, on pourra utiliser comme prototype

- |   |   |   |
|---|---|---|
| <input type="checkbox"/> 1 <code>void *f(int, int *);</code>        | <input type="checkbox"/> 2 <code>void (*f)(int , * int);</code> | <input type="checkbox"/> 3 <code>void f(int, int *);</code> |
| <input type="checkbox"/> 4 <code>void (*f)(int , int &amp;);</code> | <input type="checkbox"/> 5 <code>void (*f)(int , int *);</code> | <input type="checkbox"/> 6 <code>void f(int *, int);</code> |
| <input type="checkbox"/> 7 <code>void (*f)(int *, int);</code>      | <input type="checkbox"/> 8 <code>void f(int n, int *p);</code>  |   |

**Question 24.** Après les définitions `int x, t[4], *p;` parmi les expressions suivantes, lesquelles sont syntaxiquement correctes (c'est-à-dire ne provoquent pas d'erreur de compilation) ?

- |   |  |   |   |
|---|--|---|---|
| <input type="checkbox"/> 1 <code>x = t[-2]</code> | <input type="checkbox"/> 2 <code>x = t</code>  | <input type="checkbox"/> 3 <code>p = t</code>     | <input type="checkbox"/> 4 <code>t = p</code> |
| <input type="checkbox"/> 5 <code>x = -2[t]</code> | <input type="checkbox"/> 6 <code>p -= t</code> | <input type="checkbox"/> 7 <code>p = t + 2</code> |   |

**Question 25.** Après la définition suivante

```
int t[3] = {7 , 8 , 9};
```

que peut-on dire de l'expression `t[2]+t[3]`

- |  |   |  |
|--|---|--|
| <input type="checkbox"/> 1 correcte mais valeur imprévisible | <input type="checkbox"/> 2 elle vaut 5  | <input type="checkbox"/> 3 elle vaut 9 |
| <input type="checkbox"/> 4 erreur de compilation             | <input type="checkbox"/> 5 elle vaut 17 |  |

**Question 26.** Que produit la séquence de code suivante ?

```
int i = 10, j = 20, *p1, *p2;  
p1 = &i; p2 = &j; *p2 = *p1;  
printf("%d %d\n", i, j);
```

- |   |  |   |  |
|---|--|---|--|
| <input type="checkbox"/> 1 a che deux valeurs imprévisibles | <input type="checkbox"/> 2 a che 20 10 | <input type="checkbox"/> 3 a che 10 10          | <input type="checkbox"/> 4 erreur à la compilation |
| <input type="checkbox"/> 5 a che 20 20                      | <input type="checkbox"/> 6 a che 10 20 | <input type="checkbox"/> 7 erreur à l'exécution |  |

**Question 27.** Qu'a che le programme suivant ?

```
#include <stdio.h>  
int main(){  
    int i = 2;  
    if (!i)  
        i += 3;  
    i++;  
    if (i == 3)  
        i += 5;  
    i -= 2;  
    printf("%d\n", i);  
    return 0;  
}
```

- |                              |                              |                              |                              |
|------------------------------|------------------------------|------------------------------|------------------------------|
| <input type="checkbox"/> 1 3 | <input type="checkbox"/> 2 5 | <input type="checkbox"/> 3 4 | <input type="checkbox"/> 4 8 |
| <input type="checkbox"/> 5 6 | <input type="checkbox"/> 6 7 | <input type="checkbox"/> 7 9 |                              |

**Question 28.** Que produit le programme suivant ?

```
#include <stdio.h>
int main() {
    int tab[] = {0, 1, 2, 3 };
    int *i1 = tab + 1;
    int *i2 = tab + 2;
    int a = ++*i1 + *i2++;
    int b = ++*i1 + *i2--;
    printf("%d#%d", a, b);
    return 0;
}
```

- |  |  |                                      |
|--|--|--------------------------------------|
| <input type="checkbox"/> 1 a che 1                   | <input type="checkbox"/> 2 a che 4#4               | <input type="checkbox"/> 3 a che 4#5 |
| <input type="checkbox"/> 4 comportement imprévisible | <input type="checkbox"/> 5 a che 5#5               | <input type="checkbox"/> 6 a che 0   |
| <input type="checkbox"/> 7 a che 5#6                 | <input type="checkbox"/> 8 erreur à la compilation | <input type="checkbox"/> 9 a che 4#6 |

---

**Question 29.** Qu'a che l'exécution du programme suivant ?

```
#include <stdio.h>
int x = 6;
void ecrire(int x) {
    printf("%d ", x += 2);
}
int main() {
    ecrire(x++);
    printf("%d\n", x);
    return 0;
}
```

- |                                |                                |                                |                                |
|--------------------------------|--------------------------------|--------------------------------|--------------------------------|
| <input type="checkbox"/> 1 7 8 | <input type="checkbox"/> 2 7 7 | <input type="checkbox"/> 3 8 8 | <input type="checkbox"/> 4 9 8 |
| <input type="checkbox"/> 5 9 6 | <input type="checkbox"/> 6 8 7 | <input type="checkbox"/> 7 9 7 | <input type="checkbox"/> 8 9 9 |

---

**Question 30.** Que produit la séquence de code suivante ?

```
int i = 3, j = 10;
int *p1 = &i, *p2 = &j;
(*p1) += 2; (*p2) -= 3;
p1 = p2;
(*p1)++; (*p2)++;
printf("%d %d\n", i, j);
```

- |   |  |   |
|---|--|---|
| <input type="checkbox"/> 1 a che deux valeurs imprévisibles | <input type="checkbox"/> 2 a che 8 6               | <input type="checkbox"/> 3 terminaison sans rien a cher |
| <input type="checkbox"/> 4 che 9 5                          | <input type="checkbox"/> 5 a che 6 8               | <input type="checkbox"/> 6 a che 7 7                    |
| <input type="checkbox"/> 7 a che 5 9                        | <input type="checkbox"/> 8 erreur à la compilation | <input type="checkbox"/> 9 a che 8 8                    |
-

**Question 31.** Quel est l'e et du programme suivant ?

```
#include <stdio.h>
void f(int *p, int *q, int *r){
    int a, *b;
    b = q; q = r; r = b;
    a = *p; *p = *q; *q = a;
}
int main(){
    int x = 3, y = 5, z = 7;
    f(&x, &y, &z);
    printf("%d %d %d", x, y, z);
}
```

- |  |   |   |
|--|---|---|
| <input type="checkbox"/> 1 erreur à la compilation | <input type="checkbox"/> 2 a che 3 5 7          | <input type="checkbox"/> 3 a che 5 3 7          |
| <input type="checkbox"/> 4 a che 7 5 3             | <input type="checkbox"/> 5 a che 7 3 5          | <input type="checkbox"/> 6 erreur à l'exécution |
| <input type="checkbox"/> 7 a che 5 7 3             | <input type="checkbox"/> 8 a che n'importe quoi | <input type="checkbox"/> 9 a che 3 7 5          |

---

**Question 32.** Que produit le programme C suivant ?

```
#include <stdio.h>
#define f(m, n){static int i; i += n; n = m++ ; printf("%d ", i);}
int main(){
    static int i = 2;
    static int j = 3;
    while(i < 4){
        f(i, j);
        printf("%d ", i);
        i++;
        j += i;
        f(i, j);
    }
}
```

- |   |  |  |
|---|--|--|
| <input type="checkbox"/> 1 erreur à l'exécution | <input type="checkbox"/> 2 a che 3 2 6 6 3 10      | <input type="checkbox"/> 3 a che 4 2 7 7 3 11  |
| <input type="checkbox"/> 4 a che 4 2 7 11 3 22  | <input type="checkbox"/> 5 a che 3 2 9 15 3 25     | <input type="checkbox"/> 6 a che 3 2 6 9 3 10  |
| <input type="checkbox"/> 7 a che 3 2 6 6 3 16   | <input type="checkbox"/> 8 erreur à la compilation | <input type="checkbox"/> 9 a che 4 2 7 11 3 22 |
-

**Question 33.** On considère la séquence de code suivante :

```
// sizeof(int)=4, sizeof(double)=8, sizeof(int *)=4, sizeof(double *)=4
int t[3];
double *p1, *p2;
p1 = (double *) (t + 1);
p2 = (double *) t + 1;
```

Si l'instruction `printf("%p", t)` affiche `0x20a64`;

que va produire `printf("%p %p %p %p %p %x", p1, p2, t + 2, p1 + 1, p2 + 1, (int) p1 + 1);` ?

- ☐ 1 une erreur à la compilation
- ☐ 2 l'adresse de 0x20a68 0x20a6c 0x20a6c 0x20a70 0x20a74 20a69
- ☐ 3 l'adresse de 0x20a6c 0x20a68 0x20a6c 0x20a70 0x20a6c 20a6d
- ☐ 4 l'adresse de 0x20a68 0x20a6c 0x20a6c 0x20a70 0x20a70 20a6d
- ☐ 5 l'adresse de 0x20a68 0x20a6c 0x20a6c 0x20a70 0x20a74 20a6d
- ☐ 6 l'adresse de 0x20a65 0x20a65 0x20a66 0x20a66 0x20a66 20a6d
- ☐ 7 l'adresse de 0x20a68 0x20a68 0x20a6c 0x20a6c 0x20a6c 20a69
- ☐ 8 une erreur à l'exécution

---

**Question 34.** Après la séquence

```
int t[2]; t[0] = 1; t[2] = 0;
```

l'expression `*((t)+1)`

- ☐ 1 donne une erreur à la compilation
- ☐ 2 est une expression correcte syntaxiquement dont la valeur est imprévisible, voire dont l'évaluation provoque une erreur à l'exécution
- ☐ 3 est une expression correcte dont le type est `int` est la valeur est 1
- ☐ 4 est une expression correcte dont le type est `int` est la valeur est 0

---

**Question 35.** Après la séquence

```
int tab[] = {0, 10, 100, 1000}; int *t = &tab[1] ; int *p;
```

que peut-on dire de la séquence

```
p = &(t[1] + 1); printf("%d", *p);
```

- |                                 |   |                                  |  |
|---------------------------------|---|----------------------------------|--|
| <input type="checkbox"/> 1 a 10 | <input type="checkbox"/> 2 a 0                | <input type="checkbox"/> 3 a 101 | <input type="checkbox"/> 4 erreur à la compilation |
| <input type="checkbox"/> 5 a 1  | <input type="checkbox"/> 6 c'est imprévisible | <input type="checkbox"/> 7 a 100 | <input type="checkbox"/> 8 a 1000                  |
| <input type="checkbox"/> 9 a 11 | <input type="checkbox"/> 10 a 1001            |                                  |  |

**Question 36.** Soit la séquence :

```
// sizeof(short)=2, sizeof(double)=8, sizeof(double *)=4, sizeof(short *)=4
short n = 100;
short *ptr1 = &n;
double *ptr2 = (double *) &n;
printf("%p %p %p %p\n", ptr1, ptr2, ptr1+1, ptr2+1);
```

Si l'instruction `printf("%p", &n);` affiche `0xbffff846`  
que va produire `printf("%p %p %p %p", ptr1, ptr2, ptr1+1, ptr2+1);` ?

- ☐ 1 l'adresse de `0xbffff846` `0xbffff846` `0xbffff848` `0xbffff84e`
- ☐ 2 l'adresse de `0xbffff846` `0xbffff849` `0xbffff847` `0xbffff841`
- ☐ 3 une erreur à la compilation
- ☐ 4 l'adresse de `0xbffff846` `0xbffff846` `0xbffff848` `0xbffff848`
- ☐ 5 l'adresse de `0xbffff846` `0xbffff846` `0xbffff847` `0xbffff847`
- ☐ 6 l'adresse de `0xbffff846` `0xbffff849` `0xbffff848` `0xbffff84c`
- ☐ 7 une erreur à l'exécution
- ☐ 8 l'adresse de `0xbffff846` `0xbffff849` `0xbffff848` `0xbffff840`

---

**Question 37.** Après la définition `int ***p;` que peut-on dire de l'expression `**p + 1`

- ☐ 1 elle est syntaxiquement incorrecte    ☐ 2 elle est de type `int **`    ☐ 3 elle est de type `int`
  - ☐ 4 elle est de type `int *`    ☐ 5 elle est de type `int ***`
-

**Question 38.** On considère les trois fichiers sources suivants :

<pre> /* fichier q1.c */ #include &lt;stdio.h&gt; void f(), g(); int i; static int *p = &amp;i; int j = 10; int main() {     i++; j++; (*p)++;     printf("%d %d %d / ", , *p);     f();     printf("%d %d %d / ", i, j, *p);     g();     printf("%d %d %d", i, j, *p);     return 0; } </pre>	<pre> /* fichier q2.c */ #include &lt;stdio.h&gt; static int i = 20; int *p = &amp;i; extern int j; void f() {     i++; j++; (*p)++;     printf("%d %d %d / ", i, j, *p) } </pre>	<pre> /* fichier q3.c */ #include &lt;stdio.h&gt; int i; static int *p = &amp;i; static int j = 30; void g() {     i++; j++; (*p)++;     printf("%d %d %d / ", i, j, *p); } </pre>
---	---	--

On exécute la commande `gcc q1.c q2.c q3.c -o q` et si un exécutable a été construit, que produit son exécution ?

- ☐ 1 l'a chage de 1 11 21 / 23 31 23 / 1 11 23 /3 32 3 / 3 11 23
- ☐ 2 l'a chage de 2 11 2 / 22 31 22 / 2 11 2 /4 32 4 / 4 11 4
- ☐ 3 l'a chage de 1 11 21 / 23 12 23 / 1 12 23 /3 31 3 / 3 12 23
- ☐ 4 une erreur à l'exécution
- ☐ 5 une erreur à l'édition de liens
- ☐ 6 l'a chage de 2 11 2 / 22 12 22 / 2 12 2 /4 31 4 / 4 12 4
- ☐ 7 une erreur à la compilation

**Question 39.** On considère le programme suivant :

```

#include <stdio.h>
int a = 0x20, b = 0x30, c = 0x40, d = 0x50, e = 0x60, f = 0x70, g = 0x80, h = 0x90;
unsigned char *p = (unsigned char *) &g;
main() {
    printf("a:%p b:%p c:%p d:%p e:%p f:%p g:%p h:%p\n", &a, &b, &c, &d, &e, &f, &g, &h); //ligne 5
    printf("%x %x %x %x", p[0], p[1], p[2], p[3]); // ligne 6
    /***** ligne 7
    p -= 3;
    *(p) = 1;
    printf("%x %x %x %x %x %x %x %x", a, b, c, d, e, f, g, h);
    *****/ ligne 11 */
    return 1;
}

```

Son exécution a che :

```

0x2014 b:0x2018 c:0x201c d:0x2020 e:0x2024 f:0x2028 g:0x202c h:0x2030
80 0 0 0

```

On supprime les lignes 7 et 11 et on met les lignes 5 et 6 en commentaire.

Que peut-on dire du nouveau programme ?

- |   |   |
|---|---|
| <input type="checkbox"/> 1 il a che 20 30 40 50 60 70 180 90  | <input type="checkbox"/> 2 il a che 20 30 40 50 1 70 80 90    |
| <input type="checkbox"/> 3 il a che 20 30 40 50 60 1 80 90    | <input type="checkbox"/> 4 il y a une erreur à son exécution  |
| <input type="checkbox"/> 5 il y a une erreur à sa compilation | <input type="checkbox"/> 6 il a che 20 30 40 50 60 71 80 90   |
| <input type="checkbox"/> 7 il a che 20 30 40 50 60 70 1 90    | <input type="checkbox"/> 8 il a che 20 30 40 50 61 70 80 90   |
| <input type="checkbox"/> 9 il a che 20 30 40 50 60 170 80 90  | <input type="checkbox"/> 10 il a che 20 30 40 50 160 70 80 90 |

**Question 40.** Que produit le programme suivant ?

```
#include <stdio.h>
int t1[] = {2, 3};
int t2[] = {6, 7};
void f(int **p, int *q){
    int *r;
    r = *p; *p = q; q = r;
    (**p)++; (*q)++;
}
int main(){
    int *p = t2, *q = t1;
    f(&p, q);
    printf("%d %d %d %d", t1[0], t1[1], t2[0], t2[1]);
}
```

- |  |  |  |   |
|--|--|--|---|
| <input type="checkbox"/> 1 a che 3 3 7 7           | <input type="checkbox"/> 2 a che 6 7 2 3 | <input type="checkbox"/> 3 a che 7 7 3 3 | <input type="checkbox"/> 4 erreur à l'exécution |
| <input type="checkbox"/> 5 erreur à la compilation | <input type="checkbox"/> 6 a che 3 3 6 7 | <input type="checkbox"/> 7 a che 6 7 3 3 | <input type="checkbox"/> 8 a che 2 3 6 7        |
-



Numéro de sujet 75  
Numéro d'étudiant  
Nom

Prénom

Noircissez les carrés correspondant à vos choix

	r1	r2	r3	r4	r5	r6	r7	r8	r9	r10	réservé pour la correction	
											correctes	incorrectes
Question 1	1	2	3	4								
Question 2	1	2	3	4	5	6	7	8				
Question 3	1	2	3	4	5	6						
Question 4	1	2	3	4	5	6	7	8				
Question 5	1	2	3	4	5	6	7					
Question 6	1	2	3	4	5	6	7	8	9			
Question 7	1	2	3	4	5	6						
Question 8	1	2	3	4	5	6	7	8				
Question 9	1	2	3	4	5	6						
Question 10	1	2	3	4	5	6						
Question 11	1	2	3	4	5	6	7	8				
Question 12	1	2	3	4	5	6	7					
Question 13	1	2	3	4	5	6	7	8	9			
Question 14	1	2	3	4	5	6	7	8				
Question 15	1	2	3	4	5							
Question 16	1	2	3	4	5	6	7					
Question 17	1	2	3	4	5	6						
Question 18	1	2	3	4	5	6	7					
Question 19	1	2	3	4	5	6						
Question 20	1	2	3	4	5	6	7					
Question 21	1	2	3	4	5	6	7	8				
Question 22	1	2	3	4	5	6	7	8	9			
Question 23	1	2	3	4	5	6	7	8				
Question 24	1	2	3	4	5	6	7					
Question 25	1	2	3	4	5							
Question 26	1	2	3	4	5	6	7					
Question 27	1	2	3	4	5	6	7					
Question 28	1	2	3	4	5	6	7	8	9			
Question 29	1	2	3	4	5	6	7	8				
Question 30	1	2	3	4	5	6	7	8	9			
Question 31	1	2	3	4	5	6	7	8	9			
Question 32	1	2	3	4	5	6	7	8	9			
Question 33	1	2	3	4	5	6	7	8				
Question 34	1	2	3	4								
Question 35	1	2	3	4	5	6	7	8	9	a		
Question 36	1	2	3	4	5	6	7	8				
Question 37	1	2	3	4	5							
Question 38	1	2	3	4	5	6	7					
Question 39	1	2	3	4	5	6	7	8	9	a		
Question 40	1	2	3	4	5	6	7	8				