



+3/1/56+

UNIVERSITÉ NICE SOPHIA ANTIPOLIS  
POLYTECH NICE-SOPHIA

2016-2017

Nom : BEROUK Groupe: 4  
Prénom : Zaki

## Programmation Procédurale – SI3

mardi 11 octobre 2016

**Question 1:** Quelles sont les valeurs affichées par le printf suivant:

```
{int i = 2; printf("%d, %d", ++i, i++);}
```

☐ 3, 4

☐ 4, 3

☒ le résultat dépend du compilateur

☐ 4, 4

**Question 2:** En C, return permet de sortir

☐ d'une boucle

☒ d'une fonction

☐ d'un programme

**Question 3:** Qu'affiche le printf suivant:

```
{int a=7, b=2; printf("%3.1f et %3.1f", (float) (a/b), ((float) a)/b); }
```

☐ 3.0 et 3.0

☒ 3.0 et 3.5

☐ 3.5 et 3.5

☐ 3.5 et 3.5

**Question 4:** On veut définir le type tab100 comme un tableau de 100 entiers. Quelle est la définition correcte:

☐ typedef tab100 int [100];

☐ int [100] tab100;

☐ typedef int [100] tab100;

☐ int tab100[100];

☒ typedef int tab100[100];

**Question 5:** Soit l'énoncé for suivant:

```
for (a = 0; a > b; a -= 3) f(b);
```

Celui-ci est équivalent à:

☐ { a = 0; while (a>b) { a-=3; f(b);}}

☐ { a = 0; while (a>b) f(b); a-=3;}

☒ { a = 0; while (a>b) { f(b); a-=3;}}

**Question 6:** En C, break permet de sortir

☒ d'une boucle

☐ d'une fonction

☐ d'un programme

**Question 7:** Qu'affiche le printf suivant:

```
{int a=1, b=2; printf("%d et %d", a&b, a&&b); }
```

☐ 1 et 1

☐ 0 et 0

☒ 0 et 1

☐ 1 et 0

**Question 8:** Que valent les variables x, y et i après l'exécution de:

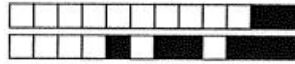
```
{int x, y, i = 5; x = i++; y = ++i;}
```

☐ 6, 6, 7

☐ 5, 6, 7

☒ 5, 7, 7

☐ 6, 7, 7



**Question 9:** Écrire la fonction `decale` qui remplace chaque caractère d'une chaîne par son caractère suivant. Ainsi, si la chaîne `a` est égale à "Hello", après l'appel `decale(a)`, `a` sera égale à "Ifmmp".

```
{ char a[10] = "Hello";  
  
  decale(a);  
  printf("%s", a); /// affiche 'Ifmmp'  
}
```

☐ 0 ☐ 1 ☐ 2 ☒ 5 Réserve au correcteur

0.5/2

```
char decale (char str[]){
```

```
  int c;  
  int i=0;
```

```
  while ( (c=getchar()) != '\0' )  
    str[i] = str[i+c];
```

```
    i++;
```

```
  return str;
```

Non, On est sur une chaîne! c'est déjà en mémoire

Non! C++ a la rigueur

! est 1 char[], la fonction

est char -

10 de fin?