

Tutorial series 01 (PW)

From 11-02-2024 to 15-02-2024

Exercise 01

Complete the empty spaces.

```
#include <stdio.h>
```

```
int main() {
    int variable = 10;
    int *ptr = &variable;

    printf("Value of variable: %d\n",....);
    printf("Address of variable: %p\n", (void*) ....);

    printf("Value using pointer: %d\n", ... );
    printf("Address stored in pointer: %p\n", (void*)....);

    *ptr = 20; // Modifier la valeur de la variable via le pointeur

    printf("Modified value using pointer: %d\n",.....);

    return 0;
}
```

Exercise 02

1. Display the detailed execution history and display of the program below:

```
# include < stdio .h >
int Toto (int *a, int *b) {
    (*a) ++;
    ++(* b) ;
    return *a+*b ;
}

void Loulou ( int *a, int *b, int *c) {
    printf (" Avant Toto : *a=%d, *b=%d, *c=%d\n", *a, *b, *c) ;
    *c = Toto (a, b) ;
}
```

```
printf (" Apres Toto : *a=%d, *b=%d, *c=%d\n", *a, *b, *c) ;  
(*a) ++;  
++(* b) ;  
}
```

```
void main () {  
int a=3 , b=4 , c=4 ;  
Loulou (&a, &b, &c) ;  
printf (" Apres Loulou ; a=%d, b=%d, c=%d\n", a, b, c) ;  
}
```

2. Deduce the role of the toto() function and that of the loulou() function.

Exercise 03

Create a function that exchanges the values of two variables using pointers.

Exercise 04

Create a function that concatenates two strings using pointers and dynamic memory allocation.

Exercise 05

```

void main(void){
    int A = 1, B = 3, C = 5;
    int *P1 = &A, *P2 = &B, *P3 = &C;
    printf("*P1=%d, *P2=%d, *P3=%d\n", *P1, *P2, *P3);
    *P2 = ++(*P3);
    printf("A=%d, B=%d, C=%d\n", A, B, C);
    *P2 = (*P3)++;
    printf("*P1=%d, *P2=%d, *P3=%d\n", *P1, *P2, *P3);
    printf("P1=%p, P2=%p, P3=%p\n", P1, P2, P3);
    *P2 = 10 - *P3++;
    printf("*P1=%d, *P2=%d, *P3=%d\n", *P1, *P2, *P3);
    printf("P1=%p, P2=%p, P3=%p\n", P1, P2, P3);
    *P3 = *P2**P3;
    printf("*P1=%d, *P2=%d, *P3=%d\n", *P1, *P2, *P3);
    *P2 = ++*P1**P3++;
    printf("*P1=%d, *P2=%d, *P3=%d\n", *P1, *P2, *P3);
    printf("P1=%p, P2=%p, P3=%p\n", P1, P2, P3);
}

```

Display the detailed execution history and display of the program below:

1. Observe and analyse the results of the instructions on lines 6, 8 and 11. Deduce the difference between the expressions $a = ++*P$, $a = (*P)++$ and $a = *P++$.
2. Justify the value of $*P3$ in line 12 by explaining what happened in the instruction just before ?
3. Deduce the origin of the value of $*P3$ in line 17.