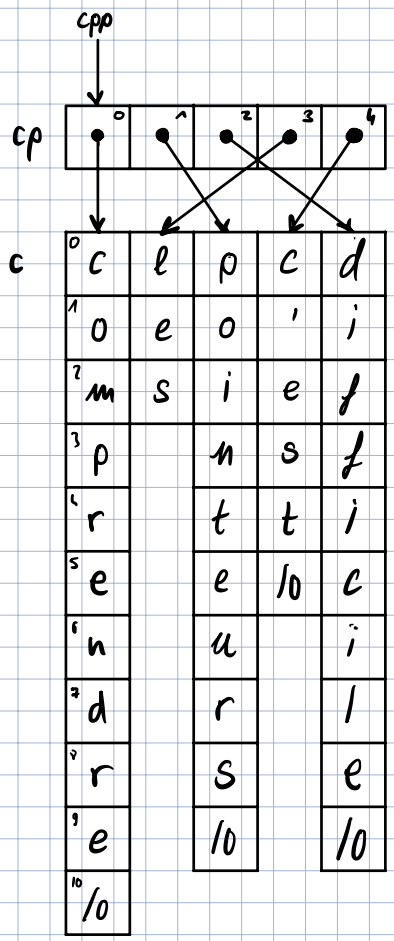



```
f(*p1, p2) {
  *p1 = 2*p2
}
```

a = 5 b = 6

int t[2] = {3, 4}

int *p = NULL, *q = NULL

f(*a, b) = 2*6 a = 12

a = 12 b = 6

1) 12, 6

p = &a q = &b

f(q, *p) = *q = 12*2

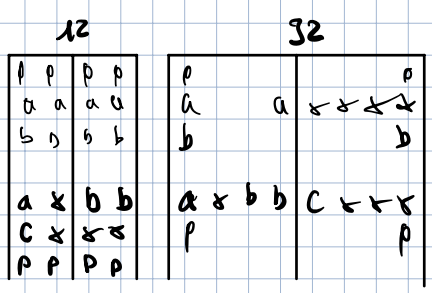
2) 12, 24 a = 12 b = 24

f(4, 3)

4 = 3*2

3) 3, 6

4) 48, 24

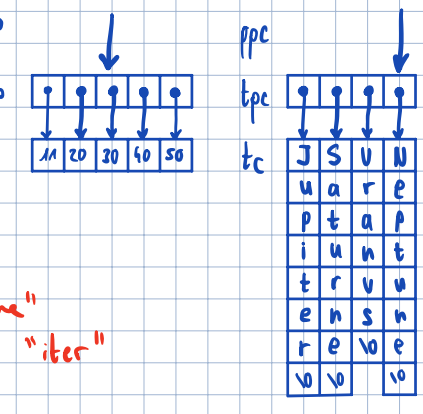


Question 1 (5 points)

Soient les déclarations suivantes :

```
int t[] = {10, 20, 30, 40, 50};
int* tp[] = {t, t+1, t+2, t+3, t+4};
int** pp = tp + 3;

const char* tc[] = {"Jupiter", "Saturne", "Uranus", "Neptune"};
const char** tpc[] = {tc, tc+1, tc+2, tc+3};
const char*** ppc = &tpc[2];
```

Quelle valeur fournit chacune des expressions ci-dessous ?
(Conseil : Aidez-vous d'un petit dessin)

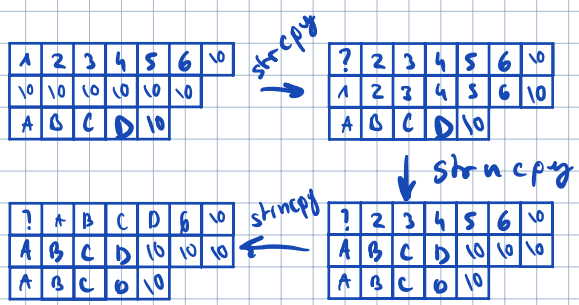


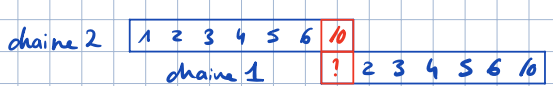
1) `tp[3][-1]` = 30 ✓
2) `**tp+(+++t)` = 10+11 = 21 ✓
3) `++++pp--` = 40 ✗ 41
4) `(--*pp)[-1]` = 11 ✓
5) `*--pp-*(tp+3)` = -2 ✓
6) `***tpc` = 'U' ✗ 'J'
7) `*++tc[3]` = 'r' ✗ 'e'
8) `*+++ppc` = "Neptun" "eptune"
9) `*--*(tpc+1)+3` = "Saturne" "iter"
10) `*++ppc[0][-2]` = 'a' ✓

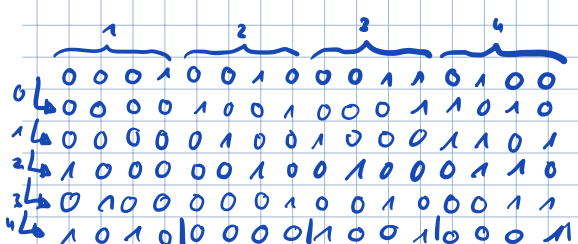
b) (2 pts)

Que va afficher à l'exécution le programme C ci-dessous ?

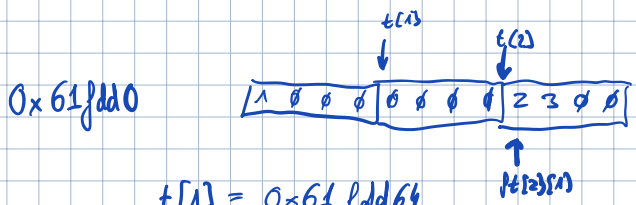
```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main(void) {
    char chaine1[] = "123456";
    char chaine2[6];
    const char* chaine3 = "ABCD";
    strcpy(chaine2, chaine1);
    strncpy(chaine2, chaine3, strlen(chaine2));
    printf("%s\n", strncpy(chaine1 + 1, chaine3, strlen(chaine2)));
    return EXIT_SUCCESS;
}
```







0x0091

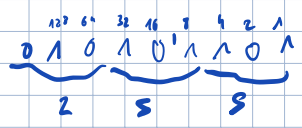


t[1] = 0x61fdd64

*(t+2) = t[2] = 0x61fdd68

*(t+2)[1] = 0x61fdd68

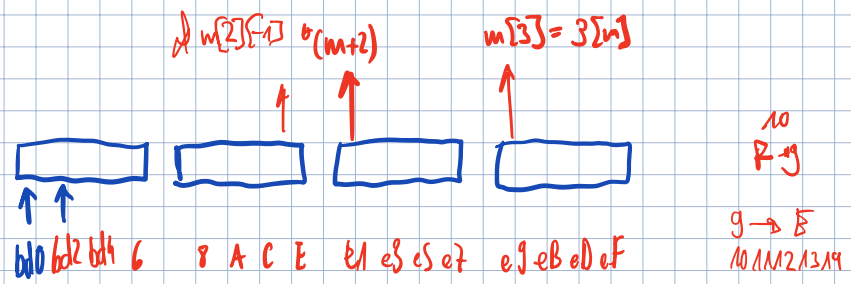
1 7 3



0x 1234, 5

Mask = 0x8000

&m[0][0] 0xc3d5fffbd0



10
R-g

g → E
10 11 12 13 14

c3d5fffbe1 + (c3d5fffbdE - c3d5fffbe9

192 → 124

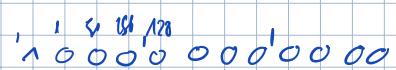
2 3 0

1 1 1 0 0 1 1 0 → 0xE6
128 64 32 16 8 4 2 1

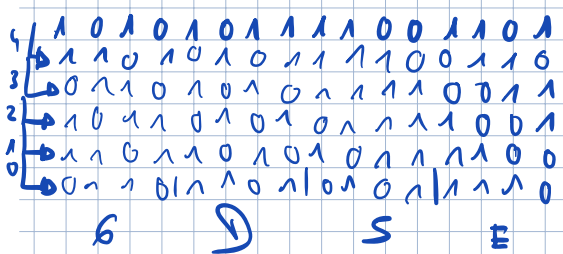
2 << 2 + 2 << 2 + 2 << 2

2 << 4 << 4 << 2

0 0 0 0 0 0 1 0



0xabcd



192

128 64 32 16 8 4 2 1

1 1 0 1 1 1 0 0

E C

D C

1010 A
1011 B
1100 C
1101 D
1110 E
1111 F

n = 1

(n+2) / 5

3.0 / 5 + 3.0 / 5

0.6 n

5^5 < 0 ? (-a) : (a)

(5 < 0) ? (-a) : (a)

5^5 = 0101 ^ 0101 = 0

0101
^ 1011
1110 → 0001 → 0010
-2

0x1234

0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 0

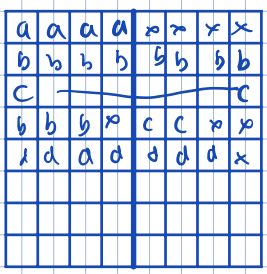
1 0 1 0 0 0 0 1 0 0 1 0 0 0 0 1

a O g ^

ddx
dex
dfx

de0 - (df0 - df4)
de0 - (-4)
de0 + 4
de4

128 64 32 16 8 4 2 1
1 0 1 0 1 1 0 1
2 6 6
A D

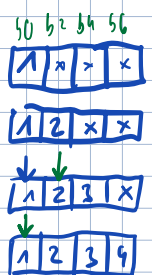


78 = 86

$$10110100 \quad 128+32+16+2 = 178$$
$$-11000000 \quad 128+64 = 192$$

$$178-192 = -14 = -E$$

$t =$



b2 - c0

b0