

CORRIGÉ

Question 1

```
int get_NIP(void){
    char ch;
    int total, valch;
    total = 0;
    do {
        ch = getch();
        if ((ch>='0') && (ch<='9')) {
            total = total*10 + ch - '0';
            printf("*");
        }
        if (ch == 27) return 0;
        if (ch == 8) {
            total = 0;
            printf("\n");
            ch = '0'; /* afin de pouvoir continuer ... */
        }
    } while ((ch>='0') && (ch<='9'));
    return total;
}
```

Question 2

Avant P: a=2 b=7
Dans P: a=8 b=3
Après P: a=3 b=7

Question 3

```
void separe( int tabA[], int tabB[], int tabC[], int dim,
             long *totB, long *totC ){
    int i;
    *totB = *totC = 0;
    for (i = 0; i < NB; i++){
        if (tabA[i] > 0) {
            tabB[i] = tabA[i];
            tabC[i] = 0;
            *totB += tabB[i];
        }
        else {
            tabB[i] = 0;
            tabC[i] = tabA[i];
            *totC += tabC[i];
        }
    }
}
```

Question 4

a=2 b=2
a=4 b=1
a=4 b=3

Question 5

```
#include <stdio.h>

double pratique( double ) ;

void main( ) ;
{ int a ; b = 5 ;
  int &pti = *a ;
  int toto[10] ;
  float x ;
  double y == 1 ;
  double *pt = &x ;

  scanf ( "%f" , &b ) ;
  printf ( "\n Valeur = ", b ) ;

  for ( j = 0 ; j <= 10 ; j++ )
    &toto [ j ] = 2( j - 1 ) ;
  scanf ( "%lf" , &y ) ;
  while ( x = y )
  {
    pratique( y ) = x ;
    getch( ) ;
  }

  /* par exemple ici, on voit bien que le bloc du while n'a pas été
     fermé. Ceci ne compte pas comme une des erreurs à trouver ! */

  printf( " le programme se termine ici " ) ;
  return 0 ;
}

void pratique( double x )
{ return x+2 ; }
```

devrait être "int"
n'a pas de type
symboles inversés: int *pti = &a;
manque le ;
== confondu avec =
les types ne correspondent pas
pt est (double *) et &x est (float *)
mauvais caractère de contrôle pour "b"
manque un caractère de contrôle
j n'a pas été déclarée
tentative de modifier l'adresse d'un emplacement
le &y devrait être à l'extérieur de la ch. de contrôle
boucle infinie
la fn retourne un double et ne peut donc
pas être utilisée à gauche d'une assignation
ne correspond pas au prototype
le ; ne doit pas apparaître ici
branchez-vous! si la fn est void, elle ne peut retourner une valeur

Question 6

```
#include <stdio.h>

int main() {
    char c;
    while ((c = getchar()) != EOF)
        putchar(c - 2);

    return 0;
}
```

Question 7

- A) Appel à swap dans « mystere » se réécrit : `swap(ptr, &y);`
La dernière assignation dans « mystere » se réécrit : `*ptr = tot;`
- B) 32

Question 8

- A)

```
int posi(double tab[], int longueur){
    int i;
    int N;
    N = 0;
    for (i = 0; i < longueur; i++)
        if (tab[i] > 0) N++;
    return N;
}
```
- B)

```
int posi2(double tab[], int longueur, double * total){
    int i;
    int N;
    N = 0;
    *total = 0;
    for (i = 0; i < longueur; i++)
        if (tab[i] > 0) { N++; *total += tab[i]; }
    return N;
}
```

Question 9 : réponse (B)

Question 10 : réponse (B)

Question 11

Il y avait plusieurs erreurs dans le code. Le code « corrigé » est :

```
#include <stdio.h>
int main() {
    int x, y;
    int * p; int * q; int * r;
    p=&x; q=&y;
    *p = 2;
    *q = 5 * *p;
    printf("%d %d\n", *p, *q);
    r = p; p = q; q = r;
    *q = x + 2;
    printf("%d %d %d\n", x, y, *r);
    return 0;
}
```

Ce code affiche :

```
2 10
4 10 4
```