Correction de l'examen d'algorithmique

Question 1.
$11001\rfloor_2 = 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0\rfloor_{10} = 16 + 8 + 1\rfloor_{10} = 25\rfloor_{10}$
$11001\rfloor_2 = 1\ 1001\rfloor_2$ et $1\rfloor_2 = 1\rfloor_{16}$ et $1001\rfloor_2 = 9\rfloor_{16}$ donc $11001\rfloor_2 = 19\rfloor_{16}$
$11001011\rfloor_2 = 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^3 + 1 \times 2^1 + 1 \times 2^0\rfloor_{10} = 128 + 64 + 8 + 2 + 1\rfloor_{10} = 203\rfloor_{10}$
$1100\ 1011\rfloor_2 = CB\rfloor_{16}$ car $1100\rfloor_2 = 12\rfloor_{10} = C\rfloor_{16}$ et $1011\rfloor_2 = 11\rfloor_{10} = B\rfloor_{16}$.

Question 2.
$87\rfloor_{10} = 1010111\rfloor_2$ donc $-87\rfloor_{10} = 10101000\rfloor_2 + 1\rfloor_2 = 10101001\rfloor_2$
$14\rfloor_{10} = 1110\rfloor_2$

Question 3.
$126\rfloor_{10} = 1111110\rfloor_2$ et $13\rfloor_{10} = 1101\rfloor_2$

```
  1111
  0001101
+ 1111110
---------
=10001011
```

Question 4.
On suppose que les tableaux sont numérotés à partir de l'indice 1.

```
doublons(t[]: tableau d'entiers, n : entier) : entier
  i,j : entiers
  répéter pour i de 1 à n
    répéter pour j de i+1 à n
      si t[i]=t[j] alors
        renvoyer 1

  renvoyer 0
```

Question 5.
On suppose que les tableaux sont numérotés à partir de l'indice 1.

```
vectoriel(u[],v[] : tableaux de réels) : tableau de réels
  w[1..3] : tableau de réels
  w[1]=u[2]*v[3]-u[3]*v[2]
  w[2]=u[3]*v[1]-u[1]*v[3]
  w[3]=u[1]*v[2]-u[2]*v[1]
  renvoyer w[]
```

Question 6.
Les valeurs écrites sont 0, 5, 4 et 20.

Question 7.
L'affichage consiste en un affichage suffixé, avec les parenthèses placés autour de chaque sous-arbre, si celui-ci n'est pas une feuille et n'est pas vide.

```
polonaise_inv(a: pointeur noeud)
    si a≠nul alors
        si (*a).gauche=nul et (*a).droite=nul alors
            écrire((*a).val)
            écrire(" ")
        sinon
            écrire("(")
            polonaise_inv((*a).gauche)
            polonaise_inv((*a).droite)
            écrire((*a).val)
            écrire(")")
```

Dans l'exemple donné, cet algorithme affichera très exactement (les espaces sont symbolisés par des ␣) : `(((8␣5␣x)(10␣(6␣2␣/)-)+)`

Question 8.
Le nombre d'arêtes est donné par la somme des coefficients de la partie supérieure de la matrice (diagonale comprise). On suppose que l'indice des tableaux commence à 1.

```
nombre_aretes(m[][]: tableau d'entiers, n : entier) : entier
    i,j : entiers
    s : entier
    s←0
    répéter pour i de 1 à n
        répéter pour j de i à n
            s←s+ m[i][j]

    renvoyer s
```