TP 2 - Corrigé

Exercice 1

```
VARIABLES

-Formule 1 EST_DU_TYPE NOMBRE
-Formule 2 EST_DU_TYPE NOMBRE
-Prix EST_DU_TYPE NOMBRE

-Prix EST_DU_TYPE NOMBRE

- DEBUT_ALGORITHME

- AFFICHER "Nombre de Formules 1:"
-- LIRE Formule 1
-- AFFICHER "Nombre de Formules 2:"
-- LIRE Formule 2
-- Prix PREND_LA_VALEUR 12 * Formule 1 + 16 * Formule 2
-- AFFICHER "Le prix à payer est : "
-- AFFICHER Prix
-- AFFICHER "€."

-- FIN_ALGORITHME
```

Exercice 2

```
■ VARIABLES

□ poids EST_DU_TYPE NOMBRE
□ taille EST_DU_TYPE NOMBRE
□ IMC EST_DU_TYPE NOMBRE
■ DEBUT_ALGORITHME
□ AFFICHER "Veuillez saisir votre poids en kilogrammes : "
□ LIRE poids
□ AFFICHER "Veuillez saisir votre taille en mètres : "
□ LIRE taille
□ //On arrondit avec la fonction round pour avoir un IMC entier
□ IMC PREND_LA_VALEUR round(poids/(taille*taille))
□ AFFICHER "Votre IMC est "
□ AFFICHER IMC
□ FIN_ALGORITHME
```

Exercice 3

```
▼ VARIABLES
    -Nbre_verres EST_DU_TYPE NOMBRE
     poids EST_DU_TYPE NOMBRE
     sexe EST_DU_TYPE NOMBRE
   taux_alcool EST_DU_TYPE NOMBRE
DEBUT ALGORITHME
    -AFFICHER "Saisir le nombre de verres consommés :"
     -LIRE Nbre_verres
     -AFFICHER "Saisir son poids en kg:"
    -LIRE poids
     -AFFICHER "Taper 1 pour un homme ou 2 pour une femme :"
     -LIRE sexe
   ▼ SI (sexe==1) ALORS
       -DEBUT_SI
-taux_alcool PREND_LA_VALEUR 10 * Nbre_verres / (poids * 0.7)
       -FIN_SI
      ▼ SINON
           -DEBUT_SINON
           -taux_alcool PREND_LA_VALEUR 10 * Nbre_verres / (poids * 0.6)
           FIN_SINON
     -AFFICHER "Le taux d'alcoolémie est "
     -AFFICHER taux_alcool

▼ SI (taux_alcool>=0.5) ALORS

       -DEBUT_SI
        -AFFICHER "La personne testée ne peut pas conduire."
       -FIN_SI
      ▼ SINON
           -DEBUT_SINON
            -AFFICHER "La personne testée peut conduire."
           FIN SINON
  FIN ALGORITHME
```

Exercice 4

```
-FONCTIONS_UTILISEES
  VARIABLES
    -nombreUtilisateur EST DU TYPE NOMBRE
     -testNombreUtilisateur EST_DU_TYPE NOMBRE
   i EST_DU_TYPE NOMBRE
nombreAlgo EST_DU_TYPE NOMBRE
DEBUT_ALGORITHME
    testNombreUtilisateur PREND_LA_VALEUR 0
     -//0 est une valeur arbitraire de départ
     -nombreAlgo PREND LA VALEUR -1
     -//-1 est une valeur arbitraire par départ. De cette manière on est sur que par hasard la valeur par défaut n'est pas celle saisit par l'utilisateur
   ▼ TANT_QUE (testNombreUtilisateur == 0) FAIRE
       -DEBUT_TANT_QUE
        -AFFICHER "Choisir un nombre entre 0 et 5"
       LIRE nombreUtilisateur
       ▼ SI (nombreUtilisateur >= 0 ET nombreUtilisateur <= 5) ALORS
           -DEBUT_SI
           testNombreUtilisateur PREND_LA_VALEUR -1
           FIN_SI
       FIN_TANT_QUE
     -i PREND_LA_VALEUR 0
   TANT_QUE (nombreAlgo != nombreUtilisateur) FAIRE
       -DEBUT_TANT_QUE
        -i PREND_LA_VALEUR i+1
        -nombreAlgo PREND_LA_VALEUR round(random()*5)
       FIN_TANT_QUE
     -AFFICHER "Nombre de tentatives : "
   AFFICHER i
  FIN_ALGORITHME
```

Exercice 5

```
-nombre_de_lancers EST_DU_TYPE NOMBRE
      compteur EST_DU_TYPE NOMBRE
      trage EST_DU_TYPE NOMBRE
    tab_face EST_DU_TYPE LISTE
▼ DEBUT_ALGORITHME
    -//Initialisation des cases du tableau à 0
   ▼ POUR compteur ALLANT_DE 1 A 6
       -DEBUT POUR
       -tab_face[compteur] PREND_LA_VALEUR 0
-FIN_POUR
    //Enregistrement du nombre entré par l'utilisateur dans la variable nombre_de lancers
     -LIRE nombre_de_lancers

    //Simulation de l'expérience

   ▼ POUR compteur ALLANT_DE 1 A nombre_de_lancers
       -DEBUT_POUR
         -//La valeur tirage enregistre le numéro de la face qui sort
       -tirage PREND_LA_VALEUR floor(random()*6+1)
         -//on augmente de 1 la variable tab_face[tirage]
       -tab_face[tirage] PREND_LA_VALEUR tab_face[tirage]+1
       FIN_POUR
    -//Le programme calcule la fréquence d'apparition de chaque face en pourcentage puis l'affiche
   ▼ POUR compteur ALLANT_DE 1 A 6
        -DEBUT_POUR
        -tab_face[compteur] PREND_LA_VALEUR 100*tab_face[compteur]/nombre_de_lancers
        -AFFICHER "Fréquence en % de la face "
        -AFFICHER compteur
        -AFFICHER ":
         -AFFICHER tab_face[compteur]
         FIN POUR
  FIN_ALGORITHME
```

Exercice 6

```
-FONCTIONS_UTILISEES
 VARIABLES
     -tailleTableau EST_DU_TYPE NOMBRE
     -tableau EST_DU_TYPE LISTE
    -i EST_DU_TYPE NOMBRE
     -variableTableau EST_DU_TYPE NOMBRE
▼ DEBUT_ALGORITHME
    LIRE tailleTableau
   -i PREND_LA_VALEUR 0
   ▼ POUR i ALLANT_DE 0 A tailleTableau-1
        -DEBUT_POUR
        -AFFICHER "Saisir valeur tableau"
        -LIRE variableTableau
        -tableau[i] PREND_LA_VALEUR variableTableau
        -FIN_POUR
   ▼ POUR i ALLANT_DE 0 A tailleTableau-1
       -DEBUT_POUR
       -AFFICHER tableau[i]
        -FIN_POUR
 -FIN ALGORITHME
```

Exercice 7

Autre corrigé possible interactif :

https://www.xm1math.net/algobox/exemples/pgcd_euclide.html

```
▼ VARIABLES
     —dividende EST_DU_TYPE NOMBRE
     -diviseur EST_DU_TYPE NOMBRE
     —quotient EST_DU_TYPE NOMBRE
     —reste EST_DU_TYPE NOMBRE
     -var_calcul EST_DU_TYPE NOMBRE
▼ DEBUT_ALGORITHME
     -quotient PREND_LA_VALEUR 0
     -reste PREND_LA_VALEUR 0
     —LIRE dividende
     -LIRE diviseur
     -///////////////////// var_calcul sert de mémoire tampon pour stocker des calculs intermédiaires //////////////
     -var_calcul PREND_LA_VALEUR dividende
    -////////////////////// On soustrait le diviseur au dividende tant que l'on peut ///////////////////
   ▼ TANT_QUE (var_calcul >= diviseur) FAIRE
        -DEBUT_TANT_QUE
         -quotient PREND_LA_VALEUR quotient + 1
         -var_calcul PREND_LA_VALEUR var_calcul - diviseur
       FIN_TANT_QUE
     -///////////// Calcul du reste avec la formule dividende = quotient * diviseur + reste /////////////
     reste PREND_LA_VALEUR dividende - diviseur * quotient
     -AFFICHER "Le quotient est : "
     -AFFICHER quotient
     -AFFICHER "Le reste est : "
     -AFFICHER reste
  FIN_ALGORITHME
```