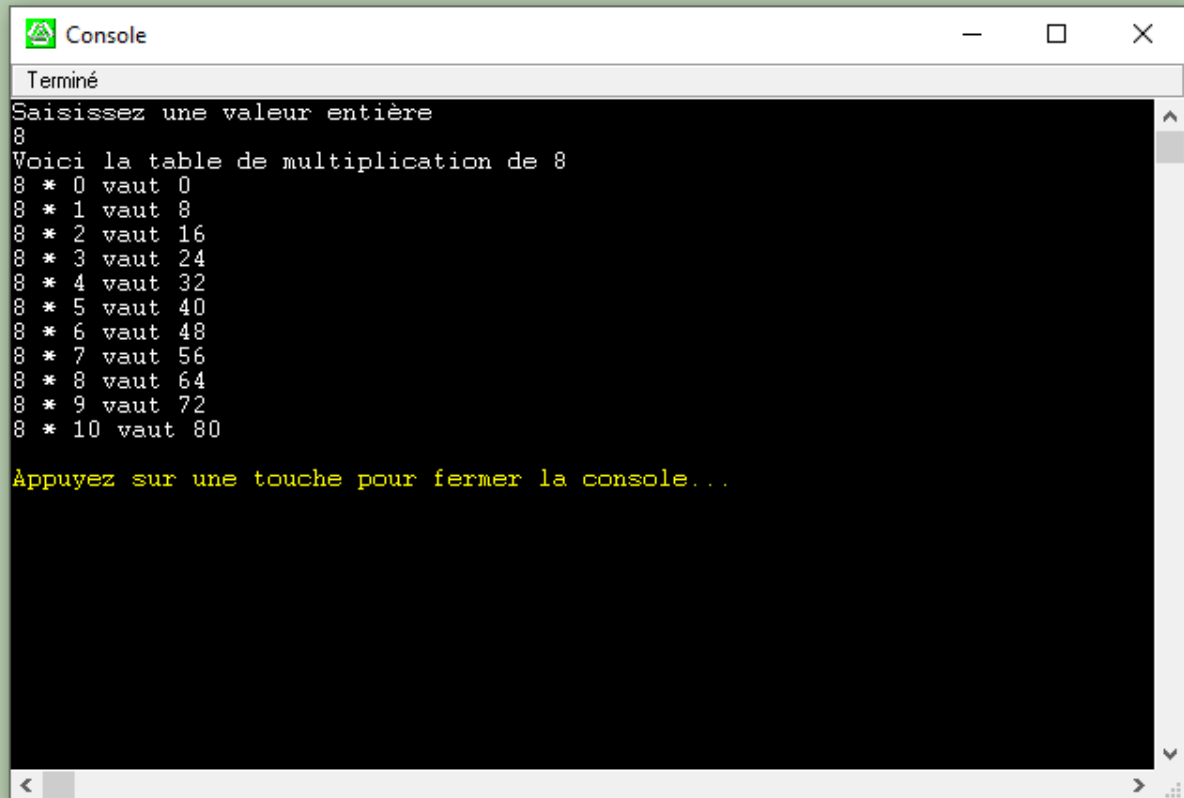


PETIT LOUIS

EXERCICE 1 : Écrire un algorithme qui demande à l'utilisateur de saisir une valeur exclusivement numérique et entière, qui calcule puis affiche la table de multiplication de celle-ci.

```
\\ Module principal
DÉBUT
    ECRIRE "Saisissez une valeur entière"
    LIRE valeur
    TANTQUE valeur!=ENTIER FAIRE
        ECRIRE "Votre valeur n'est pas un entier"
        LIRE valeur
    FINTANTQUE
    ECRIRE "Voici la table de multiplication de", valeur
    POUR i=0 jusqu'a 10 FAIRE
        ECRIRE valeur, "*", i, "vaut", valeur*i
    FINPOUR
FIN
```



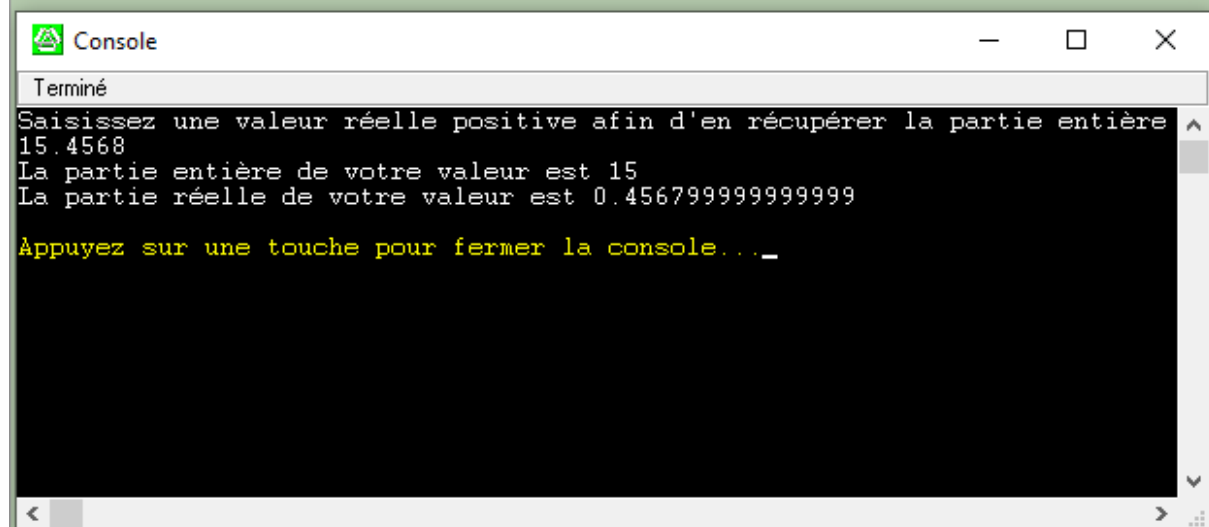
```
Console
Terminé
Saisissez une valeur entière
8
Voici la table de multiplication de 8
8 * 0 vaut 0
8 * 1 vaut 8
8 * 2 vaut 16
8 * 3 vaut 24
8 * 4 vaut 32
8 * 5 vaut 40
8 * 6 vaut 48
8 * 7 vaut 56
8 * 8 vaut 64
8 * 9 vaut 72
8 * 10 vaut 80

Appuyez sur une touche pour fermer la console...
```

PETIT LOUIS

EXERCICE 2 : Écrire un algorithme qui calcule, à l'aide d'une boucle TANTQUE, la partie entière d'un nombre réel positif précédemment saisi

```
\\ Module principal
DÉBUT
    partieEntiere=0
    partieReelle=0
    ECRIRE "Saisissez une valeur réelle positive afin d'en récupérer la partie entière"
    LIRE valeur
    TANTQUE valeur!=ENTIER ET valeur!=FLOTTANT FAIRE
        ECRIRE "Votre valeur n'est ni un entier, ni un réel"
        LIRE valeur
    FINTANTQUE
    TANTQUE valeur<0 FAIRE
        ECRIRE "Votre valeur n'est pas positive veuillez réitérer votre requête"
        LIRE valeur
    FINTANTQUE
    SI valeur=FLOTTANT ALORS
        TANTQUE partieEntiere<valeur FAIRE
            partieEntiere=partieEntiere+1
        FINTANTQUE
        SI partieEntiere!=valeur ALORS
            partieEntiere=partieEntiere-1
        FINSI
    SINON
        partieEntiere=valeur
    FINSI
    partieReelle=valeur-partieEntiere
    SI valeur>0 ALORS
        ECRIRE "La partie entière de votre valeur est", partieEntiere
        ECRIRE "La partie réelle de votre valeur est", partieReelle
    SINON
        ECRIRE "Votre valeur est nulle et donc vaut", valeur
    FINSI
FIN
```



```
Console
Terminé
Saisissez une valeur réelle positive afin d'en récupérer la partie entière
15.4568
La partie entière de votre valeur est 15
La partie réelle de votre valeur est 0.4567999999999999
Appuyez sur une touche pour fermer la console..._
```

PETIT LOUIS

EXERCICE 3 : Écrire un algorithme qui demande successivement 10 valeurs numériques à l'utilisateur, qui affiche ensuite tous les nombres saisis puis, le plus petit d'entre eux.

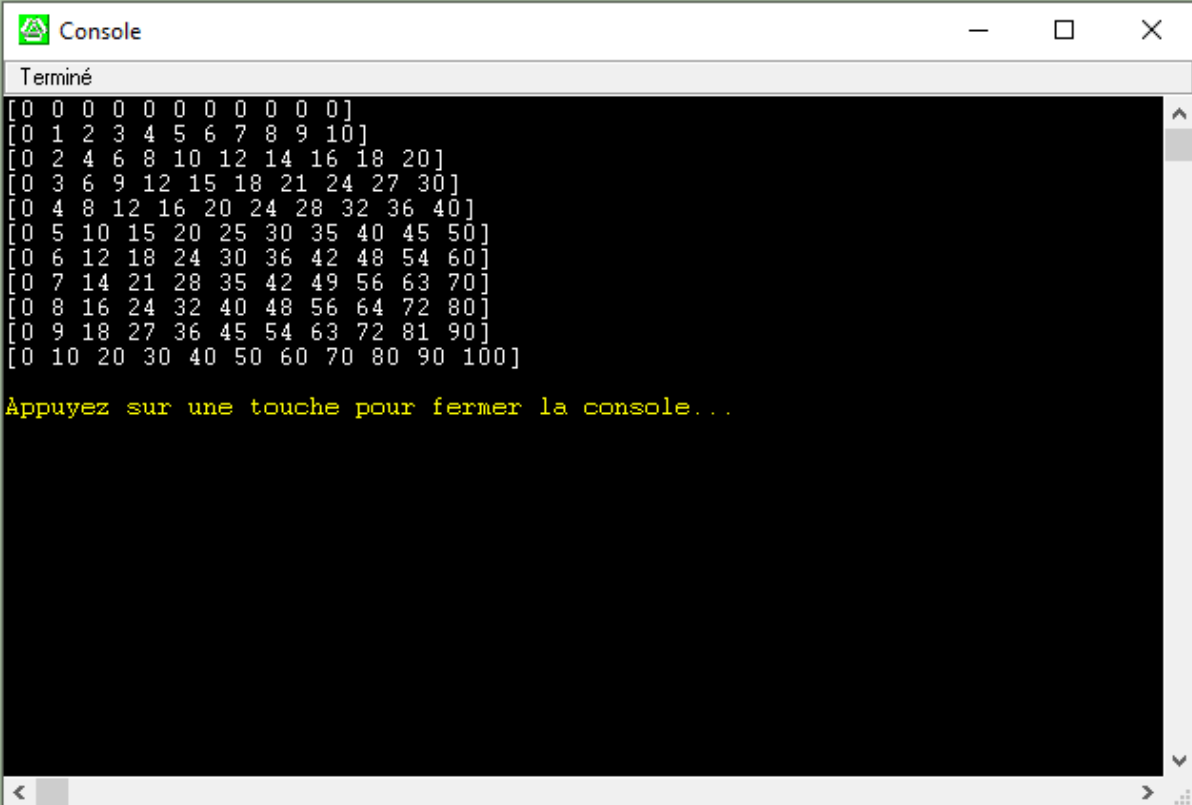
```
\\ Module principal
DÉBUT
    tableauValeur=[]
    valeurLaPlusBasse=0
    ECRIRE "Vous allez devoir saisir 10 valeurs consécutives afin d'en ressortir la valeur la plus petite"
    POUR i=1 JUSQU'A 10 FAIRE
        ECRIRE "Saisissez la valeur n°", i-1
        LIRE tableauValeur[i]
        TANTQUE tableauValeur[i]!="" ET tableauValeur[i]!="" FAIRE
            ECRIRE "Votre valeur n'est pas numérique veuillez réitérez votre saisie"
            LIRE tableauValeur[i]
        FINTANTQUE
        SI tableauValeur[i]<valeurLaPlusBasse ALORS
            valeurLaPlusBasse=tableauValeur[i]
        FINSI
    FINPOUR
    ECRIRE tableauValeur
    ECRIRE "La valeur la plus basse dans cette série de valeurs est", valeurLaPlusBasse
FIN
```

ERREUR : COMME VALEURLAPLUSBASSE INITIALISEE A 0 L'ALGORITHME RETOURNE 0 à la fin au lieu de la valeur la plus basse

PETIT LOUIS

EXERCICE 4 : Écrire un algorithme qui calcule et qui affiche à la suite les 10 premières tables de multiplication (table de zéro comprise).

```
\\ Module principal
DÉBUT
    tableMultiplication=[]
    POUR i=1 JUSQU'A 11 FAIRE
        tableMultiplication[i]=i-1
        POUR j=1 JUSQU'A 11 FAIRE
            tableMultiplication[i][j]=(i-1)*(j-1)
        FINPOUR
        ECRIRE tableMultiplication[i]
    FINPOUR
FIN
```

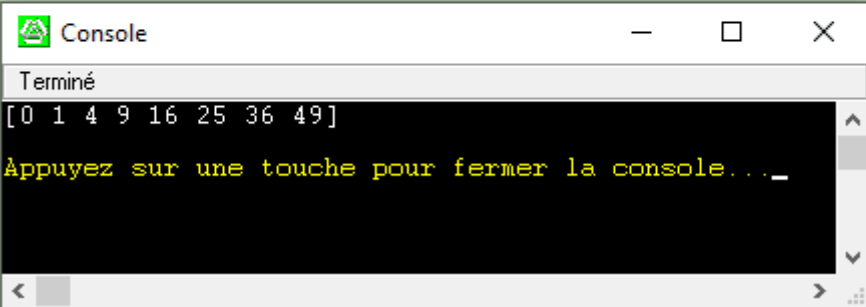


```
Terminé
[0 0 0 0 0 0 0 0 0 0 0]
[0 1 2 3 4 5 6 7 8 9 10]
[0 2 4 6 8 10 12 14 16 18 20]
[0 3 6 9 12 15 18 21 24 27 30]
[0 4 8 12 16 20 24 28 32 36 40]
[0 5 10 15 20 25 30 35 40 45 50]
[0 6 12 18 24 30 36 42 48 54 60]
[0 7 14 21 28 35 42 49 56 63 70]
[0 8 16 24 32 40 48 56 64 72 80]
[0 9 18 27 36 45 54 63 72 81 90]
[0 10 20 30 40 50 60 70 80 90 100]
Appuyez sur une touche pour fermer la console...
```

PETIT LOUIS

EXERCICE 5 : Écrire un algorithme qui boucle sur lui-même jusqu'à remplir un tableau avec huit valeurs. Chacune des cases du tableau prend pour valeur son indice multiplié par lui-même. Afficher ce tableau.

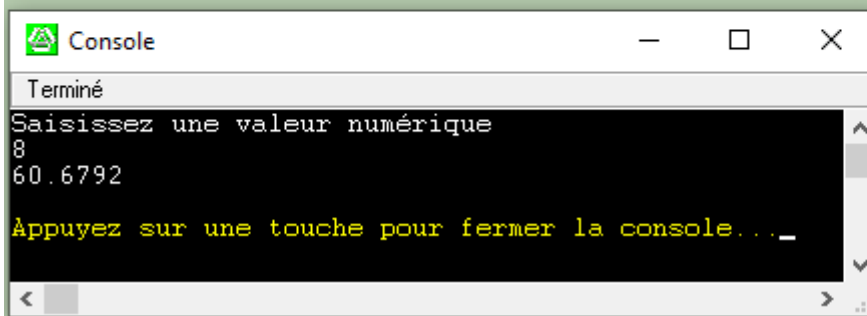
```
\\ Module principal
DÉBUT
    tableCarre=[]
    POUR i=1 JUSQU'A 8 FAIRE
        tableCarre[i]=(i-1)*(i-1)
    FINPOUR
    ECRIRE tableCarre
FIN
```



PETIT LOUIS

EXERCICE 6 : Écrire un algorithme qui demande à l'utilisateur d'exclusivement saisir une valeur numérique. Cet algorithme fait appel à une fonction créée pour l'occasion qui permet de convertir la valeur saisie en Yen. La somme convertie est finalement affichée.

```
\\ Module principal
DÉBUT
    ECRIRE "Saisissez une valeur numérique"
    LIRE valeur
    TANTQUE valeur!=ENTIER ET valeur!=FLOTTANT FAIRE
        ECRIRE "Votre valeur n'est pas numérique, veuillez réitérez votre requête"
        LIRE valeur
    FINTANTQUE
    ECRIRE EXECUTER conversion_monetaire(valeur)
FIN
```



```
\\ Module auxiliaire conversion_monetaire
ENTRER param1
    resultat=param1*7.5849
RETOURNER resultat
```

PETIT LOUIS

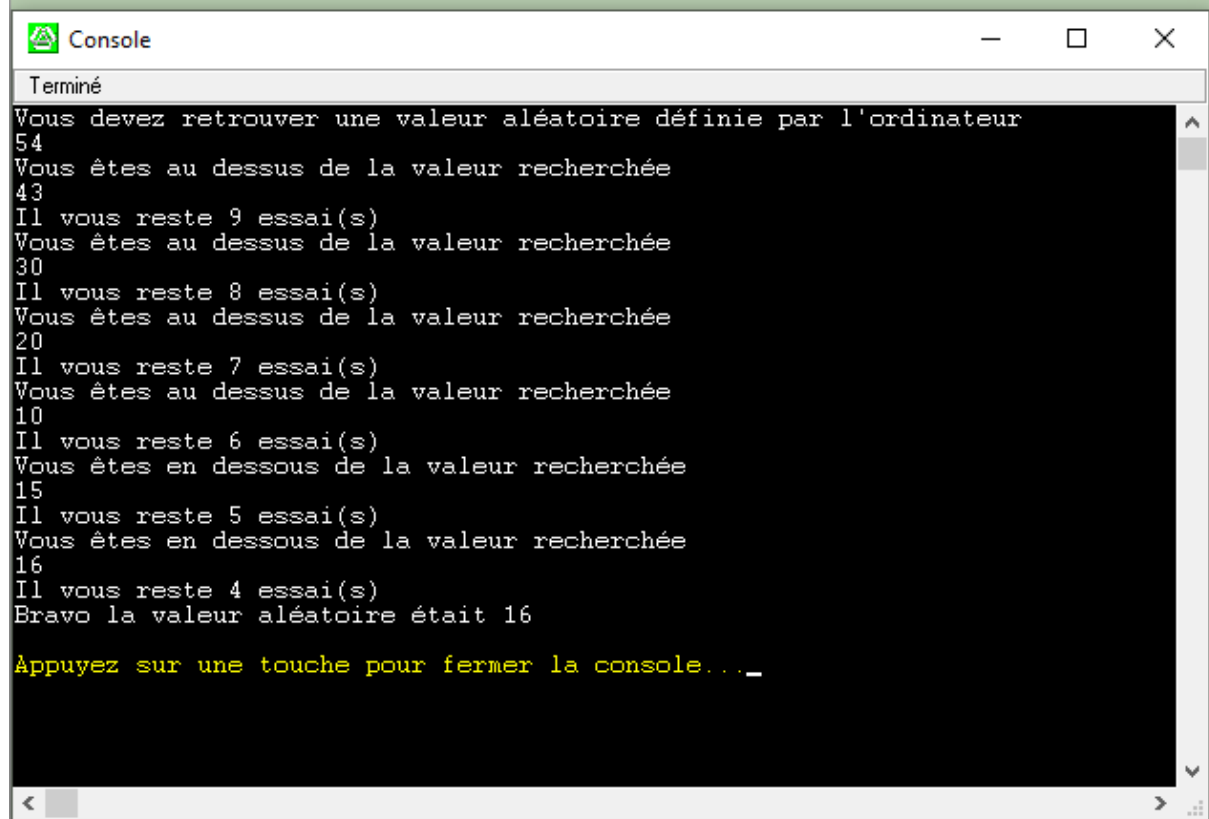
Exercice 7 : Écrire un algorithme qui demande à l'utilisateur de saisir une valeur exclusivement numérique (positive ou négative) et qui affiche ensuite le décompte de tous les chiffres entiers jusqu'à 0 (zéro compris).

```
\\ Module principal
DÉBUT
    ECRIRE "Saisissez une valeur numérique entière"
    LIRE valeur
    TANTQUE valeur!=ENTIER OU valeur=0 FAIRE
        ECRIRE "Votre valeur n'est pas un entier ou est nulle, réitérez votre saisie"
        LIRE valeur
    FINTANTQUE
    SI valeur>0 ALORS
        TANTQUE valeur!=0 FAIRE
            valeur=valeur-1
            ECRIRE valeur
        FINTANTQUE
    SINON
        TANTQUE valeur!=0 FAIRE
            valeur=valeur+1
            ECRIRE valeur
        FINTANTQUE
    FINSI
FIN
```

PETIT LOUIS

Exercice 8 : Écrire un algorithme qui fait deviner un chiffre à l'utilisateur, défini aléatoirement par l'ordinateur, en indiquant à l'utilisateur s'il est au-dessus ou au-dessous de la bonne valeur jusqu'à trouver le bon chiffre.

```
DÉBUT
    ECRIRE "Vous devez retrouver une valeur aléatoire définie par l'ordinateur"
    valeurAleat=ALEATOIRE(0,100)
    nbEssais=10
    LIRE valeurUser
    TANTQUE valeurUser!=valeurAleat ET nbEssais>0 FAIRE
        SI valeurUser<valeurAleat ALORS
            ECRIRE "Vous êtes en dessous de la valeur recherchée"
        SINON SI valeurUser>valeurAleat ALORS
            ECRIRE "Vous êtes au dessus de la valeur recherchée"
        FINSI
        LIRE valeurUser
        nbEssais=nbEssais-1
        ECRIRE "Il vous reste", nbEssais, "essai(s)"
    FINTANTQUE
    SI nbEssais=0 ET valeurUser!=valeurAleat ALORS
        ECRIRE "Vous n'avez plus d'essai disponible"
    SINON SI valeurUser=valeurAleat ALORS
        ECRIRE "Bravo la valeur aléatoire était", valeurAleat
    FINSI
FIN
```



```
Console
Terminé
Vous devez retrouver une valeur aléatoire définie par l'ordinateur
54
Vous êtes au dessus de la valeur recherchée
43
Il vous reste 9 essai(s)
Vous êtes au dessus de la valeur recherchée
30
Il vous reste 8 essai(s)
Vous êtes au dessus de la valeur recherchée
20
Il vous reste 7 essai(s)
Vous êtes au dessus de la valeur recherchée
10
Il vous reste 6 essai(s)
Vous êtes en dessous de la valeur recherchée
15
Il vous reste 5 essai(s)
Vous êtes en dessous de la valeur recherchée
16
Il vous reste 4 essai(s)
Bravo la valeur aléatoire était 16
Appuyez sur une touche pour fermer la console..._
```

ERREUR : Non prise en compte des valeurs autres que entiers pour retourner une erreur

PETIT LOUIS

Exercice 9 : L'objectif de cet exercice est de simuler une partie de "Shi Fu Mi" contre une IA.

- La partie s'effectue en 3 manches.
- Les coups possibles sont stockés au sein d'un tableau unidimensionnel tel que : coupsPossibles = ["Pierre", "Feuille", "Ciseaux"]
- Afin d'éviter la triche, l'ordinateur génère une valeur aléatoire (comprise entre 1 et 3) qui correspond à un indice du tableau précédemment déclaré en amont de l'utilisateur.
- Cette valeur n'est bien-entendu pas affichée à l'écran tant que l'utilisateur n'a pas joué son coup.
- L'utilisateur écrit le coup en toutes lettres.
- Si le coup n'est pas référencé parmi ceux définis dans le tableau "coupsPossibles", afficher "TENTATIVE DE TRICHE DE LA PART DE L'UTILISATEUR" et déclarer la manche en cours perdue par ce dernier.
- A l'issue de chaque manche, l'algorithme affiche la phrase suivante : XXXXXXXX a remporté la Xième manche.
- Les résultats de chaque manche sont stockés au sein d'un tableau.
- A l'issue des trois manches, l'algorithme retourne l'identité du vainqueur de la façon suivante : "L'ordinateur a remporté la partie."
- N'oubliez pas non plus de traiter l'éventualité selon laquelle l'IA et le joueur font le même coup! (EGALITÉ)

```
\\ Module principal
DÉBUT
  rounds=3
  coupsPossibles={"Pierre", "Feuille", "Ciseaux"}
  userChoice=""
  iaChoice=0
  resultat=0
  manches=[]
  POUR i=1 JUSQU'A rounds FAIRE
    iaChoice = ALEATOIRE(1,3)
    ECRIRE "Round",i
    ECRIRE "Pierre, Feuille ou Ciseaux ?"
    LIRE userChoice
    SI userChoice!="Pierre" ET userChoice!="Feuille" ET userChoice!="Ciseaux" ALORS
      ECRIRE "TENTATIVE DE TRICHE DE LA PART DE L'UTILISATEUR"
      manches[i]=0
    SINON SI userChoice= coupsPossibles[iaChoice] ALORS
      ECRIRE "Personne ne remporte la manche", i
      manches[i]=0.5
    SINON
      SI (userChoice="Pierre" ET iaChoice=2) OU (userChoice="Feuille" ET iaChoice=3) OU (userChoice="Ciseaux" ET iaChoice=1) ALORS
        ECRIRE "IA a remporté la manche", i, "en jouant", coupsPossibles[iaChoice]
        manches[i]=0
      SINON SI (userChoice="Pierre" ET iaChoice=3) OU (userChoice="Feuille" ET iaChoice=1) OU (userChoice="Ciseaux" ET iaChoice=2) ALORS
        ECRIRE "L'utilisateur a remporté la manche", i, "en jouant", userChoice
        manches[i]=1
      FINSI
    FINSI
  FINPOUR
  resultat=manches[1]+manches[2]+manches[3]
  ECRIRE manches
  ECRIRE "Votre résultat est", resultat
  SI resultat>1.5 ALORS
    ECRIRE "L'utilisateur a gagné la partie"
  SINON SI resultat=1.5 ALORS
    ECRIRE "Egalité"
  SINON
    ECRIRE "L'IA a remporté la partie"
  FINSI
FIN
```

PETIT LOUIS

Exercice 10 : 10. EXERCICE SUR LE TRI DE TABLEAU

- a) Écrire un algorithme qui demande à l'utilisateur de renseigner successivement 10 nombres premiers différents.
- b) Définir une fonction qui trie par ordre croissant les valeurs du tableau précédemment rempli puis qui retourne celles-ci dans l'ordre "rangé".
 - a. Intégrer cette fonction à votre algorithme et afficher les nombres du plus petit au plus grand.