

Compte Rendu du projet : Jeu du nombre caché

Partie A : « le maître du jeu est la machine »

Algorithme A1/

1/

a) Les variables utilisées par l'algorithme A1 et leur type sont :

nombresecret : int (entier)

proposition : int(entier)

b) La ligne 1 permet de affecter la valeur générer par hasard par la machine à la variable nombresecret

c) La ligne 9 est effectuée si et seulement si proposition=nombresecret

d) Le rôle de cet algorithme est de comparer la valeur générer par la machine et celle saisie par le joueur puis commenter cette comparaison.

Algorithme A2 :

Pseudo-code :

Début

nombre_secret ← hasard(1;10 000)

Afficher : « Quel est le nombre de votre choix ? »

saisir ← proposition

i ← 1

Tant que nombre_secret ≠ proposition

{

i++

Si nombre_secret < proposition alors :

Afficher : « La valeur est grande »

Sinon

Afficher : « La valeur est petite »

}

Si nombre_secret = proposition :

Afficher : « Vous avez gagné »

Afficher : « Essai(s) = »+i

Fin si

Fin

f)

Algorithme A3 :

Début

nombre_secret ← hasard(1;10 000)

Afficher : « Quel est le nombre de votre choix ? »

saisir ← proposition

i ← 1

Tant que nombre_secret ≠ proposition et i < 15

{

i++

Si nombre_secret < proposition alors :

Afficher : « La valeur est grande »

Fin Si

Sinon

Afficher : « La valeur est petite »

Fin sinon

}

Fin Tant que

Si nombre_secret = proposition :

Afficher : « Vous avez gagné »

Afficher : « Essai(s) = »+i

Fin Si

Sinon

Afficher : « Vous avez dépassé votre nombre d'essais »

Fin Sinon

Fin

Résultat :

Arbre du programme :

Node	Code range
1=Expr	15.4 - 15.26
value=Call	15.4 - 15.26
func=Name	15.4 - 15.9
id='print'	
ctx=Load	
args=[...]	
0=Constant	15.10 - 15.22
value='Essai(s) ='	
kind=None	
1=Name	15.24 - 15.25
id='i'	
ctx=Load	
keywords=[]	
orelse=[...]	
0=Expr	17.4 - 17.69
value=Call	17.4 - 17.69
func=Name	17.4 - 17.9
id='print'	
ctx=Load	
args=[...]	
0=Constant	17.10 - 17.51
value='Vous avez atteint le nombre maximal de '	
kind=None	
1=Name	17.53 - 17.54
id='i'	
ctx=Load	
2=Constant	17.56 - 17.67
value='essais ! '	
kind=None	
keywords=[]	
type_ignored=0	

Programmes en Python :

```

from random import*
nombre_secret=randint(1,5)
proposition = eval(input("Quel est le nombre de votre choix ? : "))
i=1
while nombre_secret != proposition and i<15 :
    i=i+1
    if nombre_secret<proposition :
        print("La valeur est grande")
    else :
        print("La valeur est petite")
    proposition = eval(input("Ressaissez encore une fois ! : "))
if nombre_secret == proposition :
    print("Vous avez gagné !")
    print("Essai(s) =", i)
else :
    print("Vous avez atteint le nombre maximal de ", i, "essais ! ",)

```

Partie B : « Le maître du jeu est l'utilisateur »

1)

a) La machine devrait commencer par le milieu de l'intervalle fixée par l'utilisateur.

- b) Si cette première proposition est trop petite par rapport au nombre secret, le deuxième nombre proposé doit être le milieu(entier) de ce nombre et le nombre maximal de l'intervalle fixé par l'utilisateur. Si elle est trop grande, le deuxième proposé par la machine doit être le milieu(entier) entre le premier nombre de l'intervalle par l'utilisateur et le nombre qui est proposé par la machine.
- c) Les variables de l'algorithme B1) sont : min_possible (entier(int)) , max_possible(entier (int)) et proposition (entier).
- d) le rôle de cet algorithme est de générer un intervalle dans lequel se trouve un nombre secret et de récupérer la partie entière du milieu de cet intervalle.
- e) **Algorithme final :**

Début

```
saisir min_possible
saisir max_possible
proposition ← PartieEntière((max_possible+min_possible)/2)
Afficher : proposition
saisir (type boolean) validation
Si validation = true :
    Afficher : « La proposition est trop petite »
    min_possible = proposition
    proposition ← PartieEntière((max_possible+min_possible)/2)
Fin si
Sinon si validation = false
    Afficher : « La proposition est trop grande »
    max_possible = proposition
    proposition ← PartieEntière((max_possible+min_possible)/2)
Fin sinon si
Sinon
« Bravo ! Il s'agit de la bonne valeur. »
Fin sinon
Fin
```

Résultat :

Arbre de programme :

Node	Code range
<pre> id='input' ctx=Load ▢ args=[...] ▢ 0=Constant value='La réponse est elle vérifiée (Petie/Grande ou Oui)?' kind=None keywords=[] type_comment=None ▢ or_else=[...] ▢ 0=Expr ▢ value=Call ▢ func=Name id='print' ctx=Load ▢ args=[...] ▢ 0=Constant value='Veuillez suivre les indications ! ' kind=None keywords=[] or_else=[] ▢ 7=Expr ▢ value=Call ▢ func=Name id='print' ctx=Load ▢ args=[...] ▢ 0=Constant value='Valeur trouvée. Merci de votre temps !' kind=None keywords=[] </pre>	<p>19.26 - 19.79</p> <p>21.8 - 21.52</p> <p>21.8 - 21.52</p> <p>21.8 - 21.13</p> <p>21.15 - 21.51</p> <p>23.0 - 23.48</p> <p>23.0 - 23.48</p> <p>23.0 - 23.5</p> <p>23.7 - 23.47</p>

Programmes en Python :

```

min_possible = eval(input("Saisissez le minimum de l'intervalle : "))
max_possible = eval(input("Saisissez le maximum de l'intervalle : "))
print("Veuillez retenir un nombre ( connu que par vous ) compris entre [",min_possible,",",max_possible,"]")
proposition = (max_possible+min_possible)//2
print("Laissez moi devinez, le nombre est :",proposition)
validation =input("La réponse est elle vérifiée (Petie/Grande ou Oui)?")
while validation != "Oui" :
    if validation == "Petite" :
        print (" Le nombre est trop petit ? D'accord !")
        min_possible = proposition
        proposition = (max_possible+min_possible)//2
        print("La nouvelle réponse est :",proposition)
        validation =input("La réponse est elle vérifiée (Petie/Grande ou Oui)?")
    elif validation == "Grande" :
        print ("Le nombre est trop grand ? D'accord ! ")
        max_possible = proposition
        proposition = (max_possible+min_possible)//2
        print("La nouvelle réponse est :",proposition)
        validation =input("La réponse est elle vérifiée (Petie/Grande ou Oui)?")
    else :
        print ("Veuillez suivre les indications ! ")
print ("Valeur trouvée. Merci de votre temps !")

```