LIAISON BAC PRO – BTS EN MATHEMATIQUES

**Activité : Algorithmique**

**Niveau :** Seconde, première et terminale Bac Pro

**Durée** : 2 h

|  |  |
| --- | --- |
| **Objectifs** | |
| Objectif général | **Utiliser le logiciel ALGOBOX pour effectuer une suite de tâches permettant de résoudre un problème.** |
| Connaissances | Langage de programmation. |
| Capacités mathématiques | Ecrire un programme avec Algobox pour résoudre différents problèmes : valeur numérique d’une expression, résolution d’équations, ...  Utiliser le calcul itératif et les instructions conditionnelles. |
| Attitudes transversales | Etre attentif, persévérant, patient et rigoureux. |
| Capacités cognitives | Capacité de représentation (donner du sens à un algorithme).  Flexibilité mentale (changer de cadre calculatoire). |

|  |  |
| --- | --- |
| **Déroulement** | |
| **Etape 1 :**  Présentation  **Phase magistrale**  **Support** : Prof/Tableau + vidéoprojecteur et élève/cahier | *Un algorithme permet d’exécuter une liste d’instructions les unes après les autres.*  Présenter l’interface *ALGOBOX* et un exemple d’algorithme. |
| **Etape 2**  Introduction aux structures algorithmiques de base  **Phase individuelle**  **Support** : élève/cahier + PC | Réaliser, à l’aide d’Algobox, les activités 1, 2, 3, 4, 5 et 6 (voir *ANNEXE).*  Justifier, dans chaque cas, les résultats obtenus.  ***Le but*** *est de* ***s’initier*** *aux structures algorithmiques élémentaires (affectation, calcul, entrée, sortie).*  Rédiger un algorithme permettant de résoudre une équation de type : où.  Appliquer l’algorithme précédent aux exemples : |
| **Etape 2**  Boucles et instructions conditionnelles  Alternance entre phase commune et phase individuelle | *On peut aussi programmer un algorithme qui permet d’exécuter des instructions que si une certaine condition est remplie. On utilise dans ce cas l’instruction « SI … ALORS … SINON… ».*  **-Réaliser** les activités 7 et 8 en *ANNEXE*  **-Compléter** l’algorithme de l’activité 9, afin de résoudre  **-Résoudre**: .  *Lorsqu’on connait par avance le nombre de fois que l’on veut répéter les instructions, on utilise « POUR … DE … A … ».*  **Réaliser** l’activité 10 en *ANNEXE*   *Et lorsqu’il n’est pas possible de connaître par avance le nombre de répétition nécessaire on utilise « TANT … QUE … »*  **Réaliser** l’activité 11 en *ANNEXE* |
| **Etape 3**  Pour aller plus loin | **Situation :** pour une année un individu a acheté un nombre *x* de billets « cinéma » valant 6,90 euros pièce et un nombre *y* de billets « piscines » valant 4 euros pièce, pour un total de 249,60 euros.  **L’objectif est de chercher les valeurs possibles pour *x* et *y*.**  *Attention ! x et y doivent être des entiers.*  **Proposer un algorithme ou une autre méthode afin de répondre à cet objectif.** |

|  |  |  |
| --- | --- | --- |
| Activité 1 | Activité 2 | Activité 3 |
|  |  |  |
| Après exécution de l’algorithme : | Après exécution de l’algorithme :  La variable *x* contient la valeur | Après exécution de l’algorithme :  La variable *x* contient la valeur |

ANNEXE

|  |  |  |
| --- | --- | --- |
| Activité 4 | Activité 5 | Activité 6 |
|  |  |  |
| Après exécution de l’algorithme :  *Pour* | Après exécution de l’algorithme : | Déterminer y pour |

Activité 7 : Comparer deux nombres x et y

1 VARIABLES

On cherche à générer un algorithme qui affiche le plus grand de deux nombres *x* et *y*.

Compléter les lignes 9 et 13 pour qu’il réponde à cette tâche.

2 x EST\_DU\_TYPE NOMBRE

3 y EST\_DU\_TYPE NOMBRE

4 DEBUT\_ALGORITHME

5 LIRE x

6 LIRE y

7 SI (x>y) ALORS

8 DEBUT\_SI

9 AFFICHER ………

10 FIN\_SI

11 SINON

12 DEBUT\_SINON

13 AFFICHER ……

14 FIN\_SINON

15 FIN\_ALGORITHME

Activité 8 : racine carrée d’un nombre donné

On cherche à générer un algorithme qui affiche la racine carrée d’un nombre *x* s’il est positif.

Compléter les lignes 6 et 9 pour qu’il réponde à cette tâche.

1 VARIABLES

2 x EST\_DU\_TYPE NOMBRE

3 racine EST\_DU\_TYPE NOMBRE

4 DEBUT\_ALGORITHME

5 LIRE x

6 SI (…………) ALORS

7 DEBUT\_SI

8 racine PREND\_LA\_VALEUR sqrt(x)

9 AFFICHER ……………

10 FIN\_SI

11 FIN\_ALGORITHME

Activité 9 : résoudre une équation de second degré de manière générale.

1 VARIABLES

2 a EST\_DU\_TYPE NOMBRE

3 b EST\_DU\_TYPE NOMBRE

4 c EST\_DU\_TYPE NOMBRE

5 delta EST\_DU\_TYPE NOMBRE

On cherche à générer un algorithme qui affiche les solutions d’une équation de second degré.

Compléter les lignes 20, 26 et 27 pour qu’il réponde à cette tâche.

6 x1 EST\_DU\_TYPE NOMBRE

7 x2 EST\_DU\_TYPE NOMBRE

8 DEBUT\_ALGORITHME

9 LIRE a

10 LIRE b

11 LIRE c

12 delta PREND\_LA\_VALEUR b\*b-4\*a\*c

13 SI (delta<0) ALORS

14 DEBUT\_SI

15 AFFICHER "pas de solutions réelles"

16 FIN\_SI

17 SI (delta==0) ALORS

18 DEBUT\_SI

19 AFFICHER "Il y a une solution réelle double x1="

20 x1 PREND\_LA\_VALEUR …………………………

21 AFFICHER x1

22 FIN\_SI

23 SI (delta>0) ALORS

24 DEBUT\_SI

25 AFFICHER "Il y a deux solutions distinctes x1 et x2"

26 x1 PREND\_LA\_VALEUR ………………………………………

27 x2 PREND\_LA\_VALEUR ……………………………………

28 AFFICHER "première racine x1="

29 AFFICHER x1

30 AFFICHER "deuxième racine x2 = "

31 AFFICHER x2

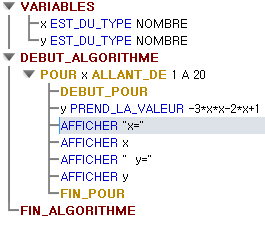
32 FIN\_SI

33 FIN\_ALGORITHME

Activité 10 : Tableau de valeurs d’une fonction

*f* est la fonction définie par : , on cherche à rédiger un algorithme afin de compléter automatiquement le tableau de valeur suivant :

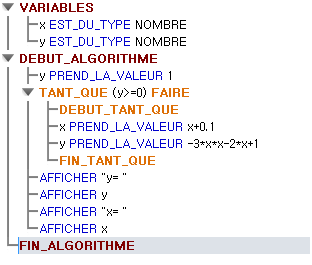
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *x* | *1* | *2* | *3* | *4* | *5* | *6* | *7* |
| *y=f(x)* |  |  |  |  |  |  |  |



1. Compléter le tableau de valeurs de *f* à l’aide de l’algorithme ci-contre.
2. Modifier cet algorithme afin de déterminer expérimentalement une valeur de *x* pour laquelle y = 0.

Activité 11 :

La fonction précédente s’annule également entre *x=0* et *x=1*.



A l’aide de l’algorithme ci-contre, déterminer la valeur de *x* pour laquelle y est nul à 0.01près.