



TP 3 : Algorithmique

Nom :

Prénom :

Note/Observation :

On considère la fonction racine carrée :

$$f(x) : \begin{matrix} [0; +\infty[\rightarrow \mathbb{R} \\ x \mapsto \sqrt{x}. \end{matrix}$$

Aide : définir une fonction, tracer son graphe, construire une table

- Définir la fonction dans $[f(x)]$: taper dans la ligne de Y_1 les commandes $[2nde] [x^2] [X,t,\theta,n]$.
- Définir les paramètres du graphe : $[fen\hat{e}tre]$ sur la ligne Xmin taper $[0]$ $[entrer]$ sur la ligne Xmax $[1]$ $[entrer]$ Xgrad permet de définir l'échelle de l'abscisse mettre 0.1 ($[0]$ $[.]$ $[1]$) $[entrer]$. Enfin en Ymin mettre 0 en Ymax mettre 1 en Ygrad mettre 0.5.
- Tracer le graphe en appuyant simplement sur $[graphe]$.
- Définir la table : mettre dans $débTbl$ la première valeur de la table ici $[0]$ et dans Pas le pas de la table c'est-à-dire la distance entre deux points d'abscisse dont on va calculer les images, ici 0.1.
- Afficher la table $[2nde] [graphe]$: la première colonne X retourne les abscisses et la deuxième colonne les ordonnées, les images des abscisses par la fonction définie par Y_1 .

Partie 1 : observation de la fonction racine carrée

- Tracer à la calculatrice la fonction f sur l'intervalle $[0; 1]$ (voir l'aide ci-dessus) puis conjecturer sa monotonie (est-elle croissante? décroissante? ni croissante ni décroissante?).
.....
- Construire à la calculatrice la table de f entre $[0; 1]$ au pas de 0,1 (voir l'aide ci-dessus). Appeler l'enseignant pour vérification.
- Compléter les lignes 2 et 3 du tableau ci-dessous. On arrondira les valeurs au centième.

x	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
$f(x)$											
$f(x + 0,1)$											×
i	0	1	2	3	4	5	6	7	8	9	×
d											×



4. Comparer $f(x)$ et $f(x + 0,1)$ et faire le lien avec la question 1.

.....
.....
.....

5. Compléter :

- « si f est une fonction croissante sur $[0, 1]$, alors pour tout $x \in \{0; 0,1; 0,2; 0,3; 0,4; 0,5; 0,6; 0,7; 0,8; 0,9\}$ on a $f(x) \dots\dots\dots f(x + 0,1)$ ».
- « s'il existe au moins un $x \in \{0; 0,1; 0,2; 0,3; 0,4; 0,5; 0,6; 0,7; 0,8; 0,9\}$ tel que $f(x) > f(x + 0,1)$ alors ».

Partie 2 : comprendre l'algorithme

On considère l'algorithme suivant :

```

x ← 0
d ← 1
Pour i allant 0 à 9
  Si f(x + 0,1) < f(x)
    alors
      d ← 0
  fin de si
  x ← x + 0,1
fin de pour
Si d = 0
  alors
    afficher « La fonction f ..... sur [0; 1] »
  sinon
    afficher « La fonction f semble ..... sur [0; 1] »
fin de si

```

6. Quelles sont les variables utilisées par ce programme ?

.....

7. Pour chaque valeur du compteur i , le programme ci-dessus va changer ou non la valeur de d . Compléter la cinquième ligne du tableau de la question 2.

8. Compléter l'algorithme.

9. Quel est le rôle de cet algorithme ?

.....
.....

10. Qu'affichera le programme si la fonction f est la fonction racine carrée ?

.....



Partie 3 : réaliser l'algorithme

1. Créer un nouveau programme : `[prgm] [NOUV] [entrer]`
2. Donner un nom au programme « CROISSAN » en appuyant sur les touches correspondantes en mode alpha puis `[entrer]`.
3. Entrer les paramètres : `[0] [sto →] [X,t,θ,n] [entrer]`
`[1] [sto →] puis D ([alpha] [x-1]) [entrer]`.
4. Créer la boucle : `[prgm] [4 :For(] taper I ([alpha] [x2]) [,] [0] [,] [9] [)] [entrer]`.
5. Créer l'instruction conditionnelle : `[prgm] [1 :If]` puis afin de faire appel à la fonction racine carrée on tape `Y1` de la façon suivante : `[var] [Y-VARS] [1 :Fonction...] [1 :Y1]`. On l'évalue en $x + 1$: `[([[X,t,θ,n] [+] [1] [)]]`. Enfin on la compare à la valeur de la fonction en x : `[2nde] [math] [5 :<]` taper `Y1` (`[var] [Y-VARS] [1 :Fonction...] [1 :Y1]`) `[([[X,t,θ,n] [)]]` `[entrer]`.
6. Remplir l'instruction conditionnelle : `[prgm] [2 :Then] [entrer]`
`[0] [sto →] D ([alpha] [x-1]) [entrer]`.
7. Terminer l'instruction conditionnelle : `[prgm] [7 :End] [entrer]`.
8. Incrémenter (augmenter d'une valeur) la variable x : `[X,t,θ,n] [+] [0] [,] [1] [sto →] [X,t,θ,n]` `[entrer]`.
9. Terminer la boucle pour : `[prgm] [7 :End] [entrer]`.
10. Créer la sortie conditionnelle : `If ([prgm] [1 :If]) D ([alpha] [x-1]) = ([2nde] [math] [1 :=]) [0]` `[entrer]`.
11. Remplir la première sortie : `Then ([prgm] [2 :Then]) [entrer]`
`Disp ([prgm] [E/S] ou [I/O] [3 :Disp]) verrouiller alpha ([2nde] [alpha]) et taper "Y1 NON CROISSANTE" [entrer]`
12. Compléter la seconde sortie : `Else ([prgm] [3 :Else]) [entrer]`
`Disp ([prgm] [E/S] ou [I/O] [3 :Disp]) verrouiller alpha ([2nde] [alpha]) et taper "Y1 SEMBLE CROISSANTE" [entrer]`
13. Terminer l'instruction conditionnelle : `[prgm] [7 :End] [entrer]`.
14. Sortir du programme : `[2nde] [mode]` et l'exécuter : `[prgm] [EXEC] [entrer]`

Voici ce que l'on doit obtenir :

```
PROGRAM :CROISSAN
: 0 → X
: 1 → D
: For(I, 0, 9)
: If Y1(X + 1) < Y1(X)
: Then
: 0 → D
: End
: X + 0.1 → X
: End
: If D = 0
: Then
: Disp "Y1 NON CROISSANTE"
: Else
: Disp "Y1 SEMBLE CROISSANTE"
: End
```