

Programme à créer

Commencer par créer un programme similaire au précédent qui donne dans une liste les 10 premiers nombres de la table de 6.

Ensuite, créer 2 sous-programmes tels que lorsque la liste de 6 est affichée, 2 actions sont possibles :

- Lorsque l'on appuie sur la flèche haut, on obtient la table de 12 en supprimant tous les nombres qui sont inutiles (on ne refait pas une liste, on supprime des éléments de la table de 6).
- Lorsque l'on appuie sur la flèche bas, on obtient la table de 3 en ajoutant tous les nombres qui manquent (on ne refait pas une liste, on complète la table de 6).

Conseil : il est plus simple de supprimer ou d'ajouter des éléments en commençant par la fin de la liste.

```

quand [drapeau] est cliqué
supprimer tous les éléments de la liste Table
mettre n à 1
répéter jusqu'à ce que n > 10
  ajouter 6 * n à Table
  ajouter 1 à n
stop ce script
    
```

```

quand la touche [flèche haut] est pressée
mettre n à 9
répéter jusqu'à ce que n = 0
  supprimer l'élément n de Table
  ajouter -2 à n
stop tout
    
```

```

quand la touche [flèche bas] est pressée
mettre n à 10
répéter jusqu'à ce que n = 0
  insérer 6 * n - 3 en position n de Table
  ajouter -1 à n
stop tout
    
```

Le problème

Il s'agit de créer un programme qui permet de ranger dans l'ordre croissant 3 nombres distincts qui seront saisis par l'utilisateur.

Solution

```

quand est cliqué
  Saisie_des_3_nombres
  Déterminer_le_plus_petit_nombre
  Déterminer_le_plus_grand_nombre
  Déterminer_le_nombre_intermédiaire
  dire regrouper le_plus_petit et regrouper < et regrouper celui_du_milieu et regrouper < et le_plus_grand pendant 5 secondes
  stop ce script
  
```

```

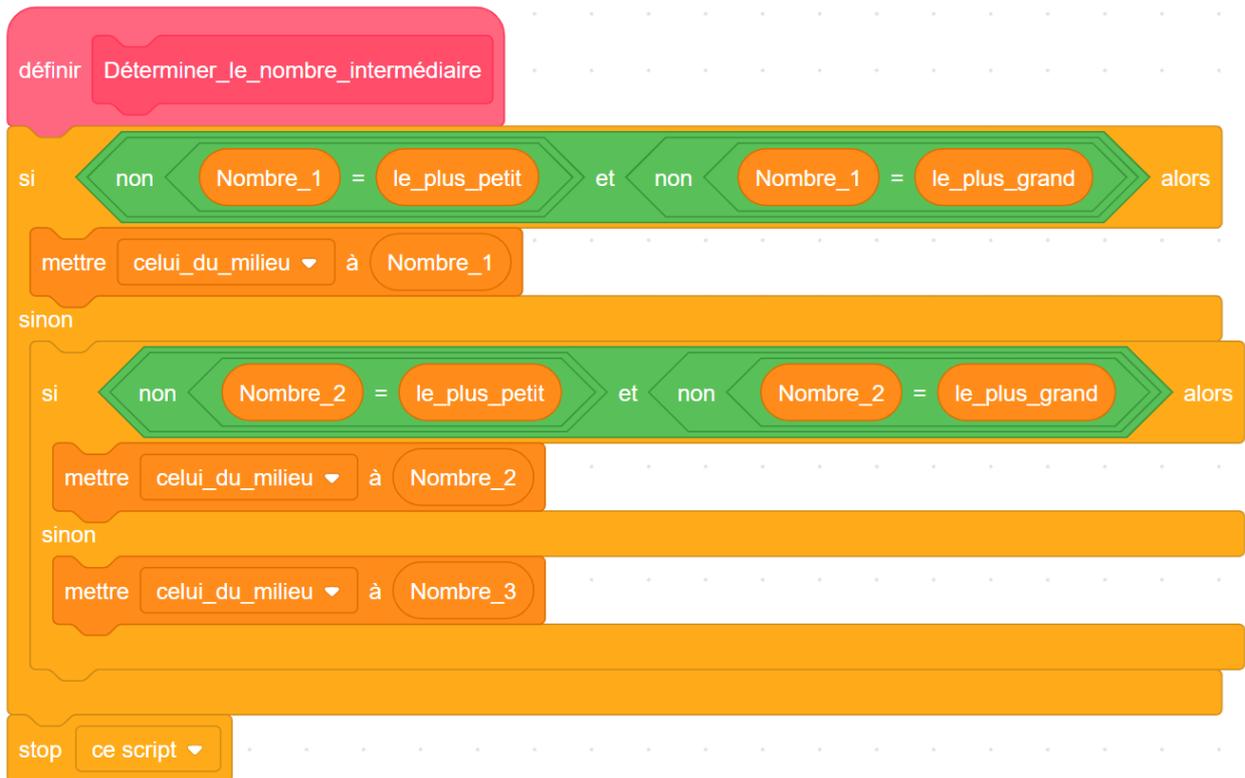
définir Saisie_des_3_nombres
  demander Quel est le premier nombre ? et attendre
  mettre Nombre_1 à réponse
  demander Quel est le deuxième nombre ? et attendre
  mettre Nombre_2 à réponse
  demander Quel est le troisième nombre ? et attendre
  mettre Nombre_3 à réponse
  stop ce script
  
```

```

définir Déterminer_le_plus_petit_nombre
  si Nombre_1 < Nombre_2 alors
    mettre le_plus_petit à Nombre_1
  sinon
    mettre le_plus_petit à Nombre_2
  si Nombre_3 < le_plus_petit alors
    mettre le_plus_petit à Nombre_3
  stop ce script
  
```

```

définir Déterminer_le_plus_grand_nombre
  si Nombre_1 > Nombre_2 alors
    mettre le_plus_grand à Nombre_1
  sinon
    mettre le_plus_grand à Nombre_2
  si Nombre_3 > le_plus_grand alors
    mettre le_plus_grand à Nombre_3
  stop ce script
  
```



Le problème

Il s'agit de créer un programme qui permet de ranger dans l'ordre croissant 3 nombres distincts qui seront saisis par l'utilisateur.

Solution

```

quand [drapeau] est cliqué
  définir Saisie_des_3_nombres
  demander "Quel est le premier nombre ?" et attendre
  mettre Nombre_1 à réponse
  demander "Quel est le deuxième nombre ?" et attendre
  mettre Nombre_2 à réponse
  demander "Quel est le troisième nombre ?" et attendre
  mettre Nombre_3 à réponse
  stop ce script

dire regrouper Nombre_1 et regrouper < et regrouper Nombre_2 et regrouper < et Nombre_3 pendant 5 secondes
stop tout
  
```

```

définir Ranger les 2 premiers nombres
  si Nombre_1 > Nombre_2 alors
    mettre Permutation à Nombre_1
    mettre Nombre_1 à Nombre_2
    mettre Nombre_2 à Permutation
  stop ce script
  
```

```

définir Ranger les 2 derniers nombres
  si Nombre_2 > Nombre_3 alors
    mettre Permutation à Nombre_2
    mettre Nombre_2 à Nombre_3
    mettre Nombre_3 à Permutation
  stop ce script
  
```

Le problème

Il s'agit de créer un programme qui permet de ranger dans l'ordre croissant 3 nombres distincts qui seront saisis par l'utilisateur. Ce programme utilise une liste.

Solution

The solution is implemented in two Scratch scripts:

- Script 1: Ranger les 3 nombres**
 - Starts with a 'when green flag is clicked' event.
 - Calls the 'Saisie des 3 nombres' script.
 - Calls the 'Ranger les 3 nombres' script.
 - Ends with a 'stop all' block.
- Script 2: Saisie des 3 nombres**
 - Starts with a 'when green flag is clicked' event.
 - Clears the 'Liste_de_nombres' list.
 - Asks for the first number and adds it to the list.
 - Asks for the second number and adds it to the list.
 - Asks for the third number and adds it to the list.
 - Ends with a 'stop all' block.

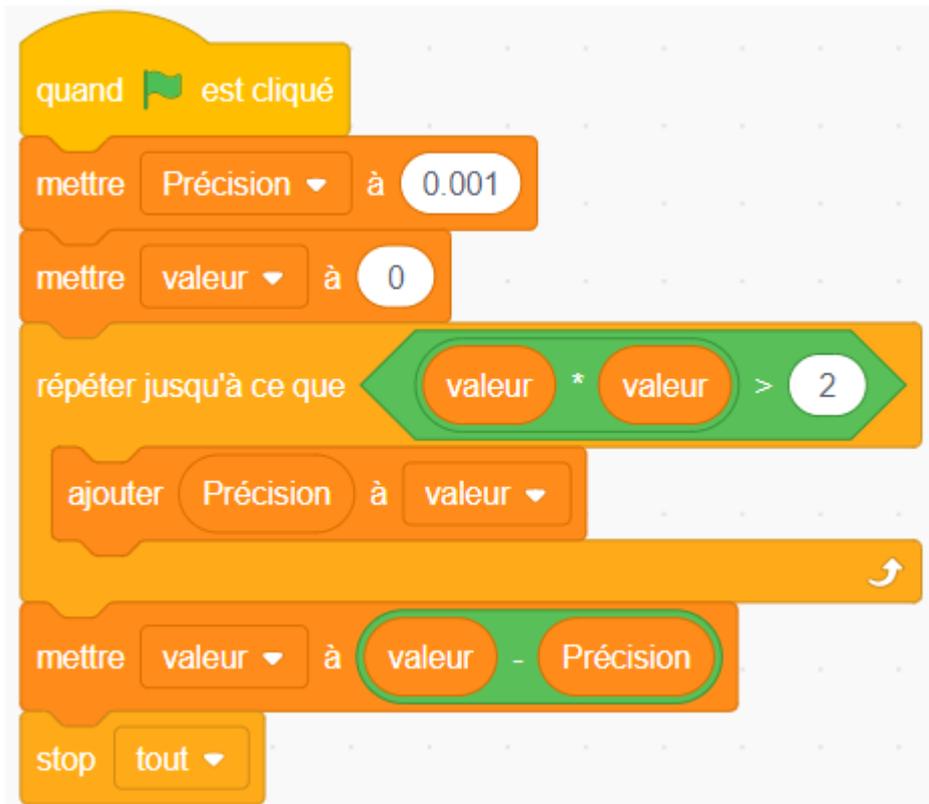
The 'Ranger les 3 nombres' script uses a bubble sort algorithm:

- It repeats a process 2 times.
- Inside the first repeat, it sets 'Rang' to 1.
- It then repeats 'jusqu'à ce que' (until) 'Rang = 3'.
- Inside this loop, it checks if the element at 'Rang' is greater than the element at 'Rang + 1'.
- If true, it swaps the two elements: 'insérer élément Rang de Liste_de_nombres en position Rang + 2' and 'supprimer l'élément Rang de Liste_de_nombres'.
- After the swap, it increments 'Rang' by 1.
- After the 'until' loop, it increments 'Rang' by 1.
- Finally, it stops the script.

Le problème

Il s'agit de créer un programme qui calcule une valeur approchée par défaut de $\sqrt{2}$ avec une précision donnée.

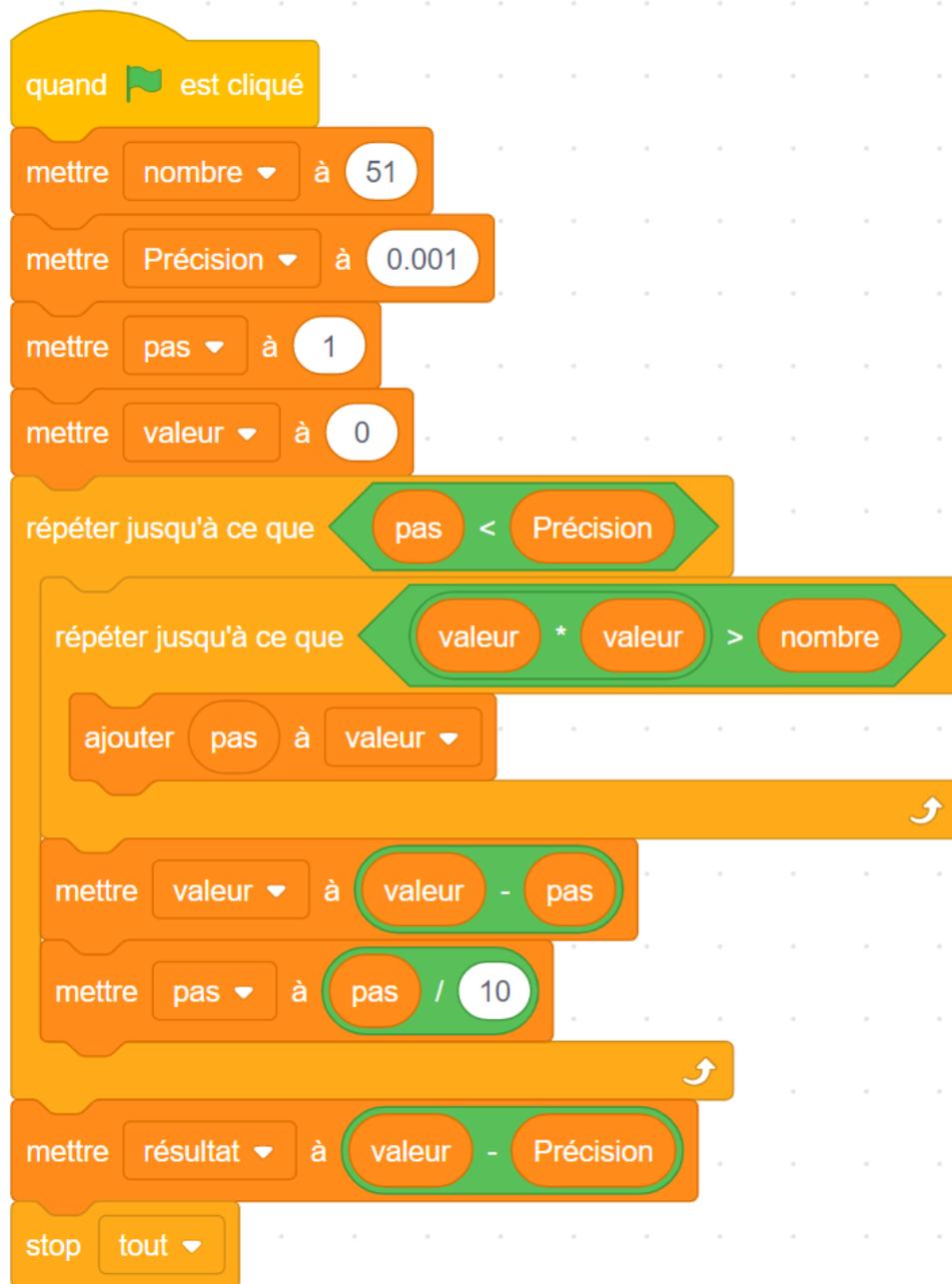
Solution



Le problème

Il s'agit de créer un programme qui calcule une valeur approchée par défaut de la racine carrée d'un nombre avec une précision donnée mais en limitant le nombre de calculs effectués.

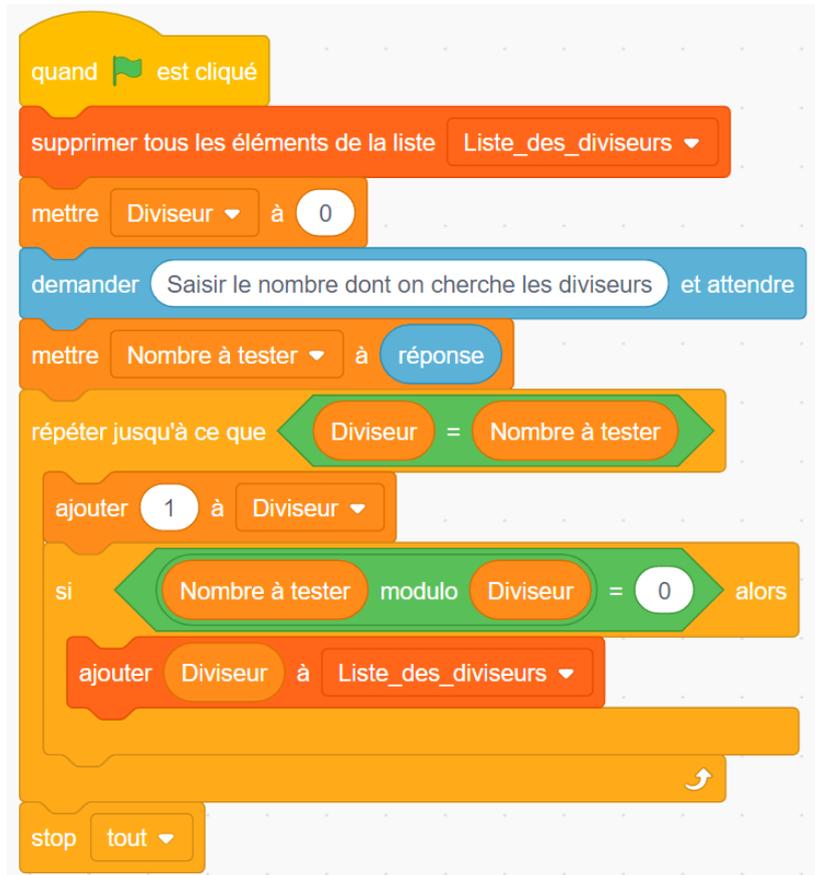
Solution



Le problème

Nous souhaitons écrire un programme qui donnera la liste complète des diviseurs d'un nombre donné.

Solution



Le problème

Nous souhaitons écrire un programme qui calcule le produit de deux nombres avec uniquement des additions. Par exemple, 6×3 peut se calculer par $6 + 6 + 6$ ou par $3 + 3 + 3 + 3 + 3 + 3$.

Solution

```

quand est cliqué
demander Saisir le premier nombre et attendre
mettre nb1 à réponse
demander Saisir le deuxième nombre et attendre
mettre nb2 à réponse
ranger nb1 nb2
produit inf sup
dire regrouper nb1 et regrouper x et regrouper nb2 et regrouper = et produit pendant 5 secondes
stop tout

```

```

définir ranger nombre1 nombre2
si nombre1 < nombre2 alors
mettre inf à nombre1
mettre sup à nombre2
sinon
mettre inf à nombre2
mettre sup à nombre1
stop ce script

```

```

définir produit facteur1 facteur2
mettre produit à 0
répéter facteur1 fois
mettre produit à produit + facteur2
stop ce script

```