

CONSTRUCTION D'UN EXERCISEUR « MULTIPLICATION PAR 10^n » UTILISANT UN CHAMP-TEXTE

Tutoriel
GeoGebra

Laurence LE FOLL

PRÉAMBULE

Généralités

Tous nos tutoriels permettant de créer des exercices utilisent la **version 5** de GeoGebra.

Dans chacun de ces tutoriels, vous trouverez :

- situé dans la partie droite de la feuille, l'objectif qui est expliqué dans le paragraphe en regard, ce qui vous permettra de retrouver facilement les manipulations à effectuer en cas de besoin ;
- un renvoi vers un autre tutoriel vous permettant d'ajouter un score à l'activité ;
- un renvoi vers un autre tutoriel vous permettant de fixer le nombre total de questions ;
- un renvoi vers un autre tutoriel expliquant comment insérer l'activité dans une leçon GeoGebra.

Objectifs spécifiques de ce tutoriel

On souhaite vérifier qu'un élève sait donner une réponse chiffrée à un calcul. Dans ce cadre, le fait que GeoGebra effectue les calculs est un inconvénient. En effet, si l'élève retape le calcul à effectuer dans le champ de réponse, sa proposition sera considérée comme correcte lors d'un test simple de vérification. Nous allons voir comment éviter ce problème en gérant les réponses, qui sont a priori des nombres, comme des textes.

Un autre objectif est de voir comment gérer la virgule décimale : par défaut le séparateur décimal est un point dans GeoGebra mais on pourra préférer utiliser la virgule. Là encore, il faudra ruser en traitant des textes.

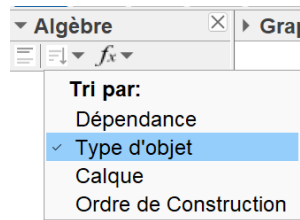
Pour illustrer ces objectifs, nous avons choisi la multiplication d'un nombre décimal par 10 ; 100 ; 1 000 ; 0,1 ; 0,01 ou 0,001.

I - AVEC LE POINT COMME SÉPARATEUR DÉCIMAL

1 - PRÉPARATION DE L'INTERFACE

Commencer par afficher la fenêtre « Algèbre » soit en cliquant sur « Affichage » puis sur « Algèbre » soit avec la combinaison de touches Ctrl + Maj + A.

Afin de retrouver plus facilement les objets qui seront construits, dérouler le menu « Algèbre », puis cliquer sur le bouton de tri  et ensuite sur « Type d'objet ».



Dans la fenêtre graphique, si les axes et/ou la grille sont visibles, les cacher en effectuant un clic-droit dans la fenêtre graphique et en cliquant sur « Axes » ou sur « Grille ».

Dans « Options », mettre « Arrondi » à 10 décimales.

2 - CRÉATION DU MULTIPLICANDE ET DU MULTIPLICATEUR ALÉATOIRES

Pour créer un nombre décimal aléatoire, appelé « nb », dont l'écriture décimale réduite comporte au maximum trois chiffres après la virgule, nous utiliserons la fonction « **AléaEntreBornes**(<Entier minimum a>,<Entier maximum b>) » qui renvoie un nombre entier aléatoire compris entre a et b.

Taper dans la barre de saisie la commande suivante.

Saisie : `nb=AléaEntreBornes(101,99999)*10^(-AléaEntreBornes(1,3))`

Créer un nombre décimal aléatoire

Créer ensuite le multiplicateur aléatoire, nommé « coeff », qui doit valoir 10 ; 100 ; 1 000 ; 0,1 ; 0,01 ou 0,001.

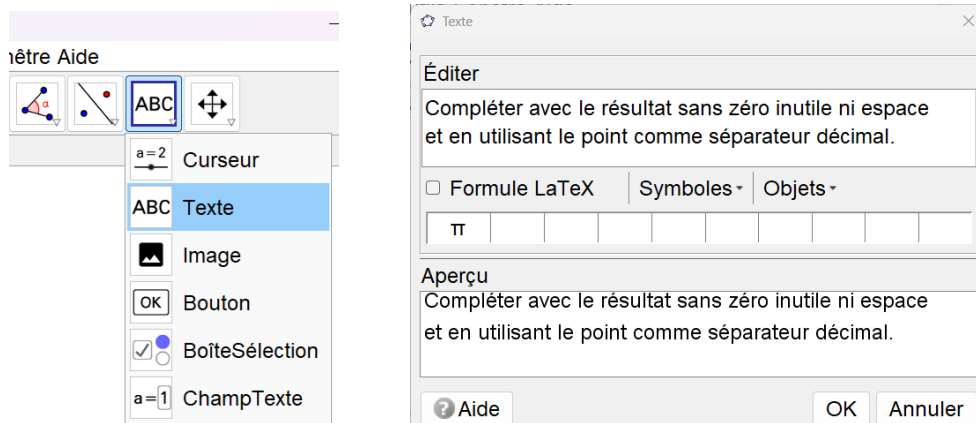
Saisie : `coeff=10^ElémentAuHasard({-3,-2,-1,1,2,3})`

On prend ici un élément au hasard dans la liste $\{-3,-2,-1,1,2,3\}$ au lieu d'utiliser la commande **AléaEntreBornes**(-3,3) afin d'éviter d'avoir un multiplicateur égal à 10^0 .

Prendre un élément au hasard dans une liste

3 - CRÉATION DE L'ÉNONCÉ

Créer un texte « Compléter avec le résultat sans zéro inutile ni espace et en utilisant le point comme séparateur décimal. ».



Valider en cliquant sur « OK ».

Dans les propriétés de texte, onglet « Texte », on peut régler sa taille sur « Grande ». Placer ce texte où vous le souhaitez dans la fenêtre graphique. Enfin rendre ce texte non sélectionnable dans l'onglet « Avancé » de ses propriétés.

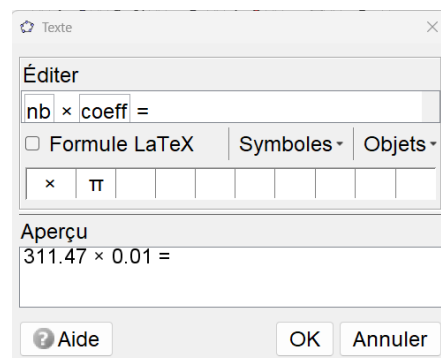
4 - CRÉATION DE LA MULTIPLICATION À EFFECTUER

Pour créer la multiplication à effectuer, nous allons utiliser un texte dynamique faisant appel aux variables « nb » et « coeff ».

Après avoir activé l'outil « Texte » et cliqué dans la zone graphique à l'endroit où vous souhaitez placer l'opération :

Créer un
texte
dynamique

- dérouler le menu « Objets » et cliquer sur « nb »,
- taper un espace,
- dérouler le menu « Symboles » puis le 2^e menu avec les opérations et cliquer sur $\times$,
- taper un espace,
- dérouler le menu « Objets » et cliquer sur « coeff »,
- taper un espace puis le signe égal,
- valider en cliquant sur « OK ».



Dans la capture d'écran ci-dessus, l'aperçu utilise les valeurs 331.47 et 0.01 respectivement pour les variables « nb » et « coeff ».

5 - CRÉATION DE LA ZONE DE RÉPONSE

Afin d'éviter qu'un calcul tapé dans la zone de réponse puisse être validé comme bonne réponse, nous allons utiliser des textes.

Commencer par créer un texte vide nommé « réponse » en tapant la commande suivante.

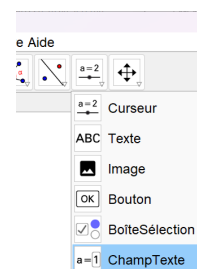
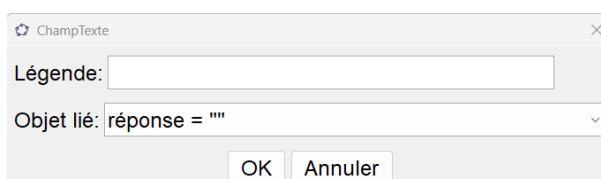
Créer un
champ-texte

Saisie : réponse=""

Cacher ensuite ce texte.

Pour créer la zone de réponse, nous allons créer un champ-texte, nommé « proposition », qui sera lié à la variable texte « réponse ». Pour cela :

- activer l'outil « Champ Texte »
- cliquer dans la fenêtre « Graphique »,
- la légende ne nous servira pas, on choisit de ne pas la renseigner,
- dans le menu « Objet lié », choisir « réponse="" »,



- valider en cliquant sur « OK ».

En liant « réponse » au champ-texte, la proposition de l'élève sera stockée dans le texte « réponse ».

Par défaut, GeoGebra n'affiche pas dans la fenêtre « Algèbre » les champs-textes qui sont des objets auxiliaires. Pour les rendre visibles, dérouler le menu de la fenêtre « Algèbre » et cliquer sur l'icône « Objets auxiliaires »  pour l'activer, elle doit être encadrée en bleu pâle : .

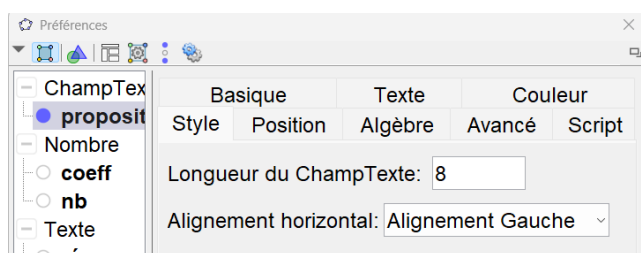
Renommer ce champ-texte « proposition ».

Effectuer un clic-droit sur « proposition » puis cliquer sur « Afficher l'étiquette » afin de masquer le nom du champ-texte dans la fenêtre graphique.

Ouvrir les propriétés du champ-texte « proposition » :

- sélectionner l'onglet « Style », dans « Longueur du ChampTexte » taper 8;
- sélectionner l'onglet « Texte » et régler la taille sur « Grande ».

Placer le champ-texte à droite du calcul affiché en maintenant le clic-droit sur le champ tout en le déplaçant. Afin de vérifier que le champ n'empiète pas sur la multiplication à effectuer, vous pouvez effectuer des tests en forçant GeoGebra à recalculer les variables aléatoires avec la combinaison de touches **Ctrl** + **R**.



Afficher les
objets
auxiliaires
dans la
fenêtre
« Algèbre »

6 - VÉRIFICATION DE LA PROPOSITION DE RÉPONSE

Créer une variable « solution » valant le produit à obtenir.

Saisie : `solution=nb*coeff`

Vérifier que les nombres affichés dans la fenêtre « Algèbre » sont en écriture décimale. S'ils sont affichés sous forme de fraction, il est conseillé de modifier leur écriture, pour cela :

- sélectionner tous les nombres en cliquant sur le mot « Nombre » dans la fenêtre « Algèbre »,
- effectuer un clic-droit et ouvrir les propriétés,
- dans l'onglet « Algèbre », décocher « Symbolique ».

Créer un booléen « correct » défini par

Saisie : `correct=réponse==" "+solution`

Transtypage

Analysons cette définition :

- la partie « " "+solution » transforme la variable nombre « solution » en texte (on réalise un transtypage),
- la partie « réponse==" "+solution »¹ constitue le test à effectuer qui compare les contenus des deux textes,

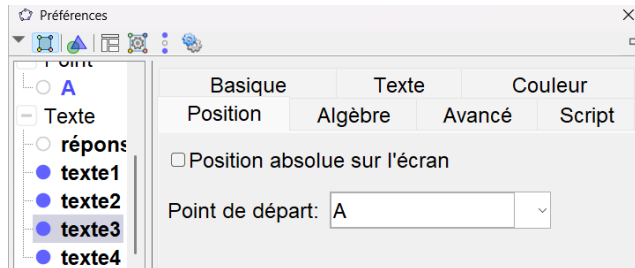
1. L'enchaînement « == » correspond à « $\frac{?}{=}$ ».

- l'ensemble du script affecte au booléen « correct » le résultat du test, ainsi « correct » vaudra « true » si la réponse proposée est le produit attendu et « false » sinon.

7 - AFFICHAGE D'UN MESSAGE DE VALIDATION

Créer deux textes « Bravo » et « Dommage ».

Placer ces deux textes au même endroit. Pour cela, créer un point A à l'endroit où les textes doivent être placés puis, dans les propriétés des deux textes, dans l'onglet « Position », choisir comme point de départ A.



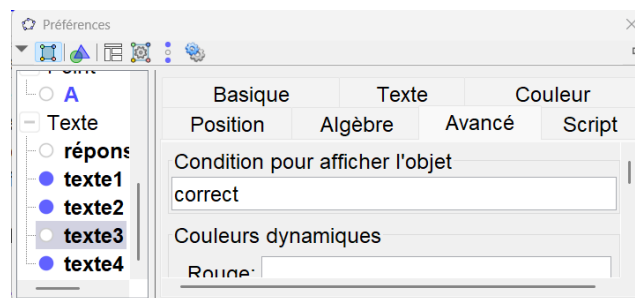
Placer plusieurs textes au même endroit

Enlever l'affichage du point A.

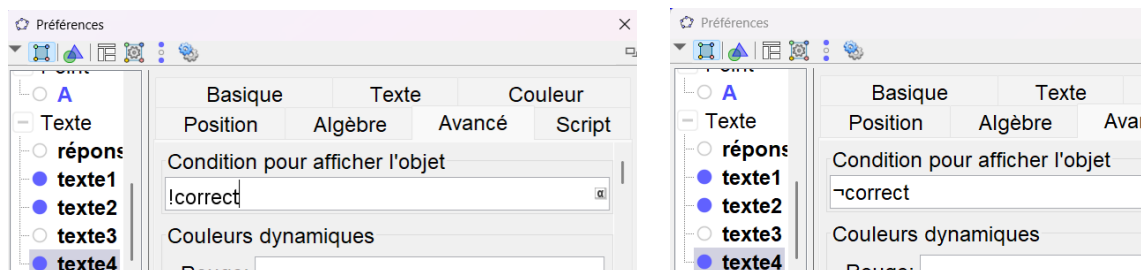
Le texte « Bravo » doit s'afficher quand la proposition est correcte.

Dans l'onglet « Avancé » des propriétés du texte, sur la ligne « Condition pour afficher l'objet », taper `correct`.

Conditionner l'affichage d'un objet



De même, le texte « Dommage » doit s'afficher quand la proposition est incorrecte. L'affichage de ce texte doit être conditionné par `!correct` que GeoGebra transformera en « $\neg$ correct »² (voir captures d'écran suivantes).



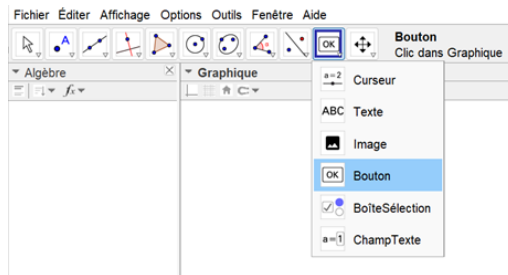
On peut changer la couleur de l'arrière-plan du texte « Bravo » en vert et celle du texte « Dommage » en orange dans les propriétés de chaque objet et régler leur taille sur « Grande ».

2. De façon générale, le point d'exclamation devant un booléen transforme celui-ci en son contraire. Il peut être apposé également devant un signe comme = ou < ou > pour signifier sa négation.

8 - INTERDIRE À L'UTILISATEUR LA CORRECTION EN CAS D'ERREUR

En l'état actuel, la validation se fait dès que l'utilisateur a écrit sa réponse dans le champ-texte. Il est alors possible à l'utilisateur de se corriger pour donner l'autre réponse en cas d'erreur.

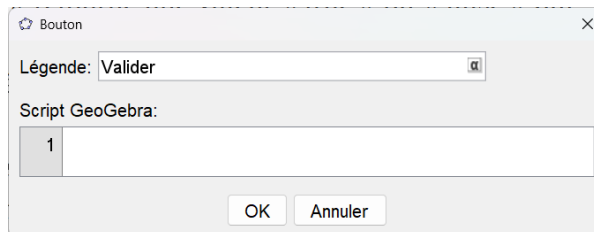
Pour éviter cela, nous allons créer un bouton permettant à l'utilisateur de demander la validation de sa proposition.



Création
d'un bouton

Une fois l'outil « Bouton » activé, cliquer dans la fenêtre « Graphique » à l'endroit où doit être placé le bouton.

Dans la légende, taper `Valider` et laisser le script vide pour l'instant.



Par défaut, GeoGebra n'affiche pas les boutons qui sont des objets auxiliaires, dans la fenêtre Algèbre. Cependant, la manipulation ayant déjà été effectuée pour afficher le champ-texte, le bouton créé doit être visible dans cette fenêtre « Algèbre ».

Ce bouton est automatiquement nommé « Bouton1 ». À l'aide d'un clic droit, le renommer « Valider ».

Ce bouton aura deux missions :

- signaler que l'utilisateur a donné et validé sa réponse pour permettre l'affichage du texte de validation,
- empêcher l'utilisateur de corriger sa réponse

Pour ce faire, nous allons créer un booléen « valid » initialisé à « false ».

Saisie : `valid=false`

Ce booléen prendra la valeur « true » lorsque l'utilisateur aura validé sa proposition de réponse.

Dans les propriétés du bouton « Valider », dans l'onglet « Script », sous-onglet « Par clic », ajouter la commande `SoitValeur(valid,true)`.

Modifier le
script d'un
bouton

Ce booléen « valid » va conditionner l'affichage des messages de validation.

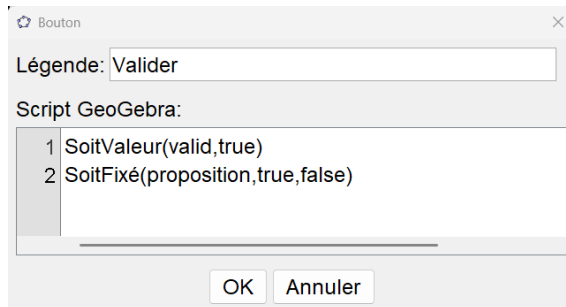
Modifier les conditions d'affichage du texte « Bravo » par `valid&&correct` et celles du texte « Dommage » par `valid&&!correct`.

Pour empêcher l'utilisateur de modifier sa réponse, nous allons utiliser la commande SoitFixé.³

Dans les propriétés du bouton, onglet « Script », ajouter la commande

`SoitFixé(proposition,true,false)`

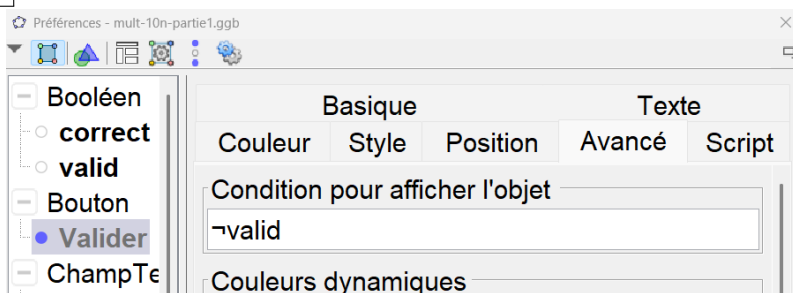
Cette commande permet de fixer la position du champ-texte (pour que l'utilisateur ne la déplace pas par mégarde) mais aussi de rendre le champ-texte non sélectionnable. L'utilisateur ne pourra donc plus changer sa réponse.



⚠ Il faut toujours fermer la fenêtre « Propriétés... » d'un bouton pour pouvoir le tester!

Il faut que le bouton « Valider » ne soit plus visible après que l'utilisateur a cliqué dessus.

Dans l'onglet « Avancé » des propriétés de ce bouton, conditionner son affichage par `!valid`.



Interdire le déplacement et la sélection d'un objet

9 - CRÉATION DU BOUTON PERMETTANT UNE AUTRE QUESTION

Créer un nouveau bouton dont la légende est « Nouveau » puis lui affecter ce script :

```
1 ActualiserConstruction()
2 SoitValeur(réponse,"")
3 SoitValeur(valid,false)
4 SoitValeur(correct,false)
5 SoitFixé(proposition,true,true)
```

Analysons ce script :

- La ligne 1, « ActualiserConstruction() » permet de recalculer les valeurs aléatoires « nb » et « coeff ».
- Dans les lignes suivantes, on réinitialise les valeurs des variables « réponse », « valid » et « correct ».

3. SoitFixé(<Objet>,<fixé true|false>,<sélectionnable true|false>) rend l'objet fixe (true) ou modifiable (false), sélectionnable (true) ou non (false).

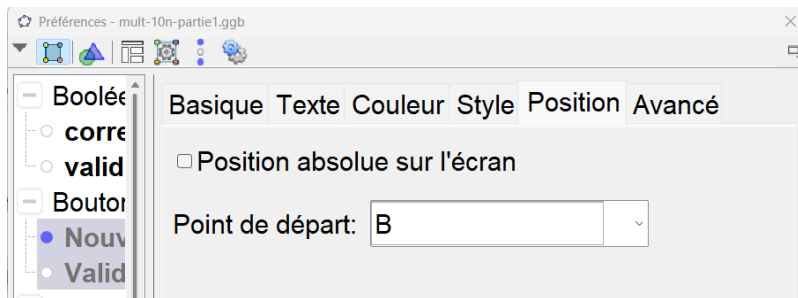
La commande « `SoitValeur(réponse, "")` » donne au texte « réponse » la valeur d'un texte vide, comme initialement.

La commande « `SoitFixé(proposition, true, true)` » permet de rendre le champ-texte « proposition » à nouveau sélectionnable et de pouvoir modifier son contenu.

Renommer ce bouton « Nouveau ».

Conditionner l'affichage de ce bouton par le booléen « valid ».

À l'aide d'un nouveau point B, positionner les deux boutons au même endroit. (Il faut décocher la case « Position absolue sur l'écran » dans l'onglet « Position » des propriétés de chaque bouton).



Cacher le point B.

10 - CRÉATION DU SCORE ET LIMITATION DU NOMBRE DE QUESTIONS

À ce stade, vous pourriez souhaiter :

- afficher un score sous la forme « nombre de réussites / nombre d'essais », pour cela vous pouvez suivre le **tutoriel dédié** ;
- fixer le nombre de questions accessibles via le bouton « Nouveau », pour cela vous pouvez suivre le **tutoriel dédié**.

11 - FINITIONS

Notre exerciceur est presque terminé.

Si besoin, replacer tous les éléments pour créer quelque-chose d'esthétique et d'ergonomique (éventuellement, pour cela, faire réafficher temporairement les points A et B).

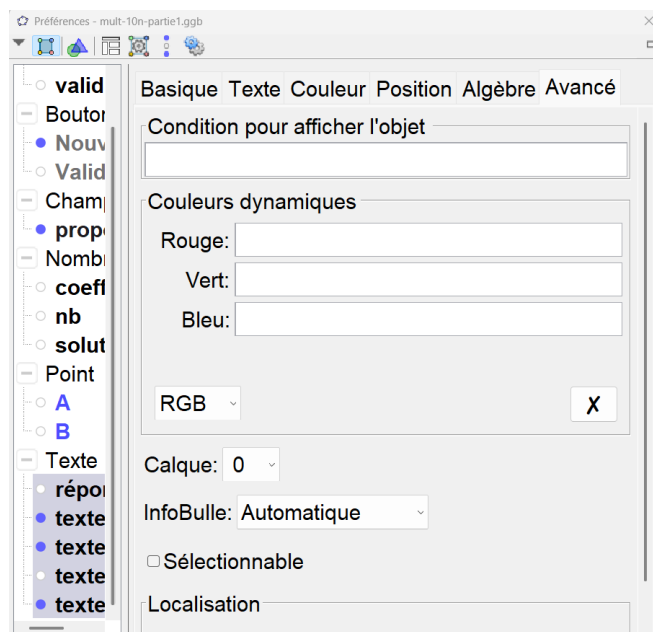
Pour éviter que l'utilisateur ne modifie ou ne supprime des éléments (volontairement ou non), il convient de rendre les textes non sélectionnables.

Sélectionner tous les textes (clic droit sur le mot « texte » dans la fenêtre « Algèbre ») puis, dans leurs propriétés, dans l'onglet « Avancé », décocher la case « Sélectionnable ».



Il est possible que cette case soit décochée mais que certains textes soient encore sélectionnables. Il est donc conseillé de cocher puis de décocher cette case pour être certain d'arriver à ses fins.

Rendre un objet non sélectionnable



Avant d'enregistrer, fermer la fenêtre Algèbre. . .
Il ne vous reste plus qu'à sauvegarder votre fichier.

12 - DONNER L'EXERCISEUR AUX ÉLÈVES

Pour cela vous pouvez **enregistrer votre activité** dans votre compte GeoGebra et **créer une leçon**.



Exercice d'application

En cas d'erreur dans la proposition, afficher la règle à utiliser en fonction du multiplicateur.

Indication : Créer 6 textes (un pour chacun des cas) et conditionner leur affichage par les valeurs de « coeff », « valid » et « correct ».

II - AVEC LA VIRGULE COMME SÉPARATEUR DÉCIMAL

On peut souhaiter utiliser la virgule comme séparateur décimal dans le multiplicande et le multiplicateur.

Reprendre le fichier précédent pour y effectuer les modifications suivantes.
Commencer par afficher la fenêtre « Algèbre » qui avait été fermée avant l'enregistrement précédent.

1 - CHANGER LE POINT EN VIRGULE

Pour transformer le point en virgule, nous allons devoir transformer les nombres en textes puis utiliser la commande « RemplacerTout(<Texte>,<Texte à rechercher>,<Texte de remplacement>) » qui recherchera dans les textes le caractère « . » et le remplacera par le caractère « , ».

Remplacer
un texte par
un autre

Créer les textes « nbaffiché », « coeffaffiché » et « solution2 » qui seront les re-transcriptions respectives de « nb », « coeff » et « solution » avec une virgule comme séparateur décimal.

Saisie : `nbaffiché=RemplacerTout(Texte(nb), ".", ",")`

Saisie : `coeffaffiché=RemplacerTout(Texte(coeff), ".", ",")`

Saisie : `solution2=RemplacerTout(Texte(solution), ".", ",")`

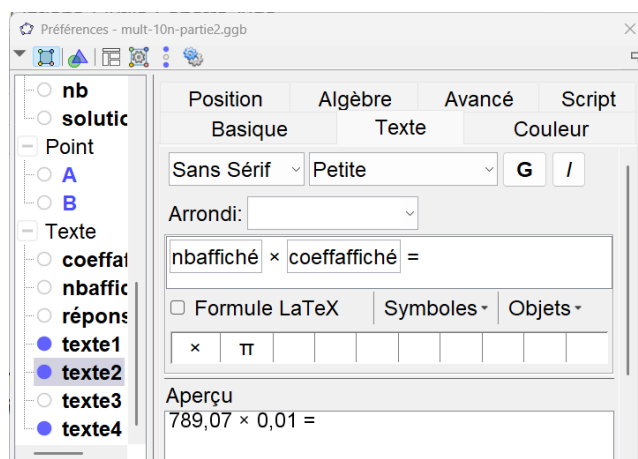
Cacher ces trois textes

2 - MODIFIER L'AFFICHAGE DE LA MULTIPLICATION

Le texte contenant la multiplication à effectuer (« texte2 ») étant non sélectionnable, accéder à ses propriétés dans la fenêtre « Algèbre ».

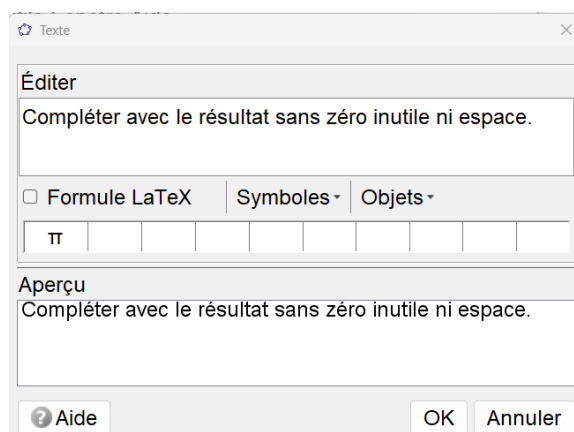
Dans l'onglet « Texte », remplacer les objets « nb » et « coeff » par « nbaffiché » et « coeffaffiché ».

Décocher « Formule LaTeX » pour éviter d'avoir un espace disgracieux après la virgule.



3 - MODIFICATION DE L'ÉNONCÉ

Afin de laisser aux élèves la possibilité d'utiliser la virgule comme séparateur décimal, modifier l'énoncé en supprimant « et en utilisant le point comme séparateur décimal ».



4 - MODIFICATION DE LA VÉRIFICATION DE LA PROPOSITION DE RÉPONSE

Désormais deux réponses sont acceptables : la solution avec un point ou avec une virgule comme séparateur décimal.

Le premier cas est celui déjà testé par le booléen « correct ».

Il nous faut ajouter un deuxième test : la variable « réponse » correspond-elle au texte « solution2 » ?

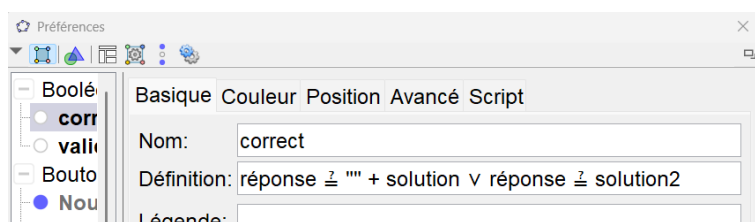
Ouvrir les propriétés du booléen « correct ».

Dans l'onglet « Basique », compléter la définition comme suit, puis appuyer sur la touche « entrée » pour valider votre saisie.

Définition : `réponse == "" + solution || réponse == solution2`

Cette définition est maintenant en deux parties séparées par le symbole « || » (obtenu en utilisant deux fois de suite la combinaison de touches Alt Gr + 6 sur un clavier français ou en insérant un symbole « V » à l'aide du bouton  à droite de la barre de définition).⁴

En fermant et rouvrant les propriétés de « correct », vous constaterez que GeoGebra a automatiquement transformé l'écriture de cette définition en :



Il ne reste plus qu'à fermer la fenêtre « Algèbre » et à enregistrer ce nouveau fichier.

4. Ces symboles « || » et « V » correspondent au connecteur logique « ou ».

INDEX DES FONCTIONS UTILISÉES

Source : <https://geogebra.github.io/docs/manual/fr/>

- **ActualiserConstruction()**
Recalcule tous les objets utilisés dans le fichier GeoGebra.
- **AléaEntreBornes(<Entier minimum a>,<Entier maximum b>)**
Retourne un nombre entier aléatoire de l'intervalle $[a; b]$.
- **Elément(<Liste>,<Position n>)**
Retourne le n^{e} élément de la liste.
- **ElémentAuHasard(<Liste>)**
Retourne un élément choisi avec une probabilité uniforme dans la liste. Tous les éléments doivent être de même type.
- **Longueur(<Liste>)**
Retourne le nombre d'éléments de la liste.
- **RemplacerTout(<Texte>,<Texte à rechercher>,<Texte de remplacement>)**
Crée un nouveau texte dans lequel les textes à rechercher ont été remplacés par les textes de remplacement.
- **Si(<Condition>,<Objet a>,<Objet b>)**
Retourne une copie de l'objet a si la condition prend la valeur true (vrai), et une copie de l'objet b si elle prend la valeur false (faux).
- **SoitFixé(<Objet>,<fixé true|false>,<sélectionnable true|false>)**
Rend l'objet fixe (true) ou modifiable (false), sélectionnable (true) ou non (false).
- **SoitValeur(<Booléen b>,<false|true>)** ou **SoitValeur(<Booléen b>,<0|1>)**
Affecte au booléen b la valeur false (0) ou true (1).
- **SoitValeur(<Liste L>,<Nombre n>,<Objet O>)**
Affecte au n^{e} élément de la liste L la valeur actuelle de l'objet O. Le nombre n est au plus égal à Longueur(L)+1.
- **Texte(<Objet>)**
Retourne la formule pour l'objet en tant qu'objet texte.