

Tuto : créer un exercice de calcul avec MathGraph32

Remarque : des tutoriels existent déjà. Ils sont ici :

<https://aide.labomep.sesamath.net/doku.php?id=contribuer:start>

Vous devez avoir la version javascript MathGraph32 6.4.7 ou supérieure, ou utiliser la version en ligne

(<https://www.mathgraph32.org/ftp/js/mtg32online/indexLyceeSansComplexes.html>)

Ce premier tutoriel est une version « minimaliste » : il permet de comprendre comment ça marche, mais n'explique pas comment, par exemple insérer un message en cas d'erreur, etc. Ceci sera fait dans une deuxième version.

1) Commencez par limiter le cadre d'exécution de la figure MathGraph32 en cliquant



sur cet icône en haut à droite :

2) Cochez 'Utiliser un cadre de dimensions données' et saisissez 750 et 568, et cliquez sur 'OK'

Options

Unité d'angle : Degré Radian

Niveau élémentaire Niveau collège

Niveau avancé sans nombres complexes Niveau avancé avec nombres complexes

Personnaliser les outils disponibles

Affichage automatique des mesures de longueur et angle

Utiliser un cadre de dimensions données

Largeur : (10 à 758) Hauteur : (10 à 568)

Coefficient multiplicateur pour l'export des images (0.25 à 4) :

[→ Elements figés](#)



3) Créez une nouvelle figure MathGraph32 : et sélectionnez 'Figure sans repère et sans longueur unité', puis cliquez sur 'OK'.

Nouvelle figure

Figure avec repère

Figure avec repère millimétré

Figure sans repère avec longueur unité

Figure sans repère et sans longueur unité

Figure de base

Figure avec repère modifiable sans vecteurs



4) Changeons la couleur de fond : puis cliquez sur l'item 'Couleur de fond de la figure' et entrez dans le champ d'édition **#f6fafa** (ceci semble anecdotique, mais, en fait, cela permet d'avoir un standard pour tous les exercices j3p de Sesamath – c'est donc important pour la charte graphique)

5) Nous allons créer un exercice qui tire au hasard trois entiers relatifs et demande de les additionner. Il y a donc *trois* nombres aléatoires à tirer. La variable de nom réservé 'nbvar' devra donc contenir la valeur 3. Créez un calcul nommé *nbvar* en utilisant



l'icone - **Attention : ce nom de variable est obligatoire** (il est utilisé par le programme pour repérer le nombre de variables aléatoires) :

6) On veut des nombres entiers entre -20 et 10 (pour avoir plus de nombres négatifs). Si $\text{rand}(0)$ est un nombre décimal entre 0 et 1 (exclu), on obtient un nombre entiers relatifs aléatoire entre -20 et 10 grâce à la formule :

-20+int(31*rand(0)). Créez trois calcul de la même façon que *nbvar* :

- un calcul *r* contenant comme formule -20+int(31*rand(0)).

- un calcul *s* contenant comme formule -20+int(31*rand(0)).

- un calcul *t* contenant comme formule -20+int(31*rand(0)).

7) Créez un calcul nommé *cal* contenant comme formule $r+s+t$ qui permettra d'afficher le texte au départ, avec les valeurs de *r*, *s* et *t* (« Calculer A=... ») :

8) Créez un calcul nommé *val/cal* contenant comme formule *cal* : cette variable contiendra le résultat de *cal*. Ce sera utile pour afficher la correction :

Calcul réel

Nom du calcul :

Formule :

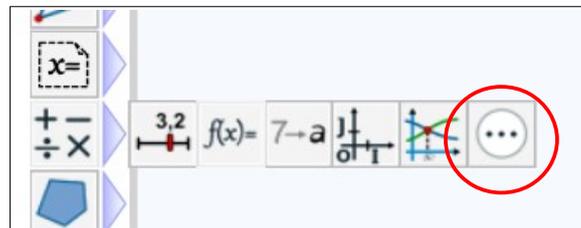
9) Créez un calcul *rep* qui contiendra la réponse de l'élève, et qui, pour l'instant, contient 0. **Attention : ce nom de variable est obligatoire** (il est utilisé par le programme pour repérer la réponse de l'élève).

Calcul réel

Nom du calcul :

Formule :

10) Il faut maintenant créer le test qui compare les contenus : cliquez sur  :



Choisissez un outil

- Renommer un calcul
- Test d'existence
- Fonction réelle de deux variables
- Fonction réelle de trois variables
- Calcul approché d'intégrale
- Test d'équivalence
- Test de factorisation
- Test d'équivalence de nature d'opérateur

Sélectionnez 'Test d'équivalence' et cliquez sur 'OK' :
Nommez le test de comparaison 'resolu'. On compare valcal et rep :

Test d'équivalence

Nom du calcul :

Choisissez les calculs à comparer :

r
s
t
cal
rep
zero
exact
valcal

nbvar
r
s
t
cal
rep
exact

Remplacement des valeurs avant la comparaison Remplacement des valeurs avant la comparaison

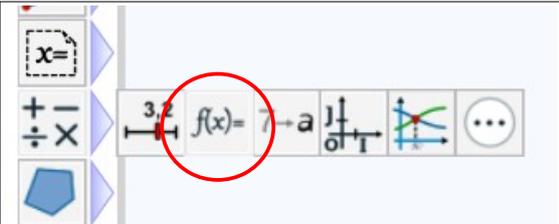
Equivalence décimal - fraction irréductible

Condition d'existence :

Validez en cliquant sur 'OK'.

11) 'resolu' contient l'information de l'égalité des expressions. Mais il ne permet pas de savoir si c'est la réponse *finale* attendue (par exemple il va détecter qu'il doit comparer $-10 = -6 - 7 + 3$ avec la réponse de l'élève $-13 + 3$, et dire qu'effectivement, c'est égal. Mais il ne détectera pas que l'élève n'a pas fourni la réponse finale, à savoir -10). Pour cela, nous créons une **fonction zero** qui donnera vraie si le calcul de la différence donne une valeur très faible, et un calcul *exact*, qui contiendra vrai si *zero* est vrai :

- Créez la fonction zero : 



Fonction

Nom de la fonction :

Variable :

Formule :

(pensez à valider en cliquant sur 'OK')

- Créez un calcul exact contenant zero(rep-cal)

Calcul réel

Nom du calcul :

Formule :

- enfin, on crée un dernier calcul *reponse* qui sera celui transmis à Labomep, et qui contiendra : 1 si l'élève a bien donné la forme la plus simple du calcul demandé, 2 si sa réponse est exacte mais n'est pas la forme demandée et 0 si elle est fausse

Attention : ce nom de variable est obligatoire (il est utilisé par le programme pour afficher la réponse de l'ordinateur) :

Calcul réel

Nom du calcul :

Formule :



12) Vérifiez que l'ordre des variables est bien celui-ci : cliquez sur , puis vérifiez que vous avez bien :

Protocole de la figure

```

pi = 3.14159265358979323846
nbvar = 3
r = -20+int(31*rand(0))
s = -20+int(31*rand(0))
t = -20+int(31*rand(0))
cal = r+s+t
valcal = cal
rep = 0
resolu
zero(x) = abs(x)<0.0000000001
exact = zero(rep-valcal)
reponse = si(resolu,1,si(exact,2,0))

```

↑↑

↑↑

↑↑

↑↑

↓↓

↓↓

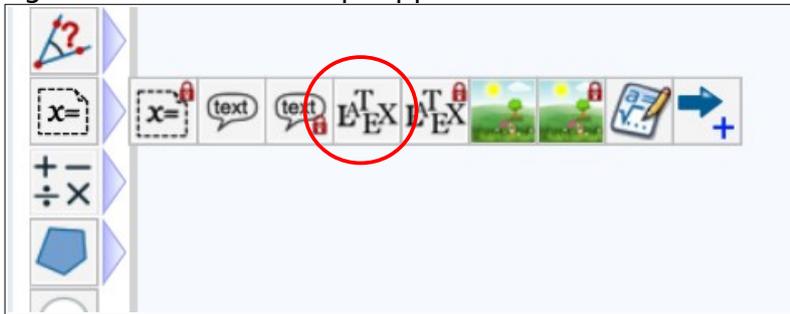
↓↓

↓↓

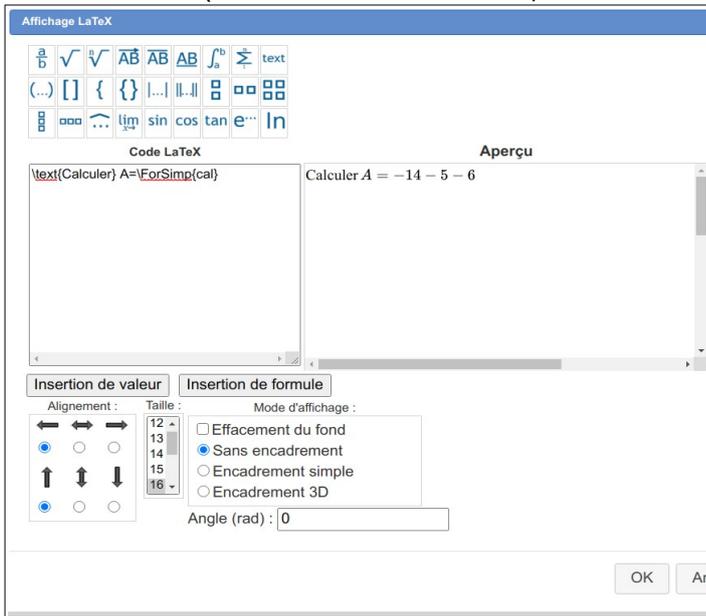
pi : Calcul
 pi = 3.14159265358979323846
 Valeur actuelle : 3.14159265359
 N°html : 0

Objets intermédiaires (*)
 Objets masqués visibles

13) Créons maintenant les **affichages** : cliquez sur , puis sur l'endroit de la figure où vous voulez qu'apparaisse l'énoncé.



Puis tapez «`\text{Calculer} A=\text{ForSimp}{cal}`» dans la fenêtre de code Latex. Vous obtenez ceci (avec d'autres valeurs, évidemment) :

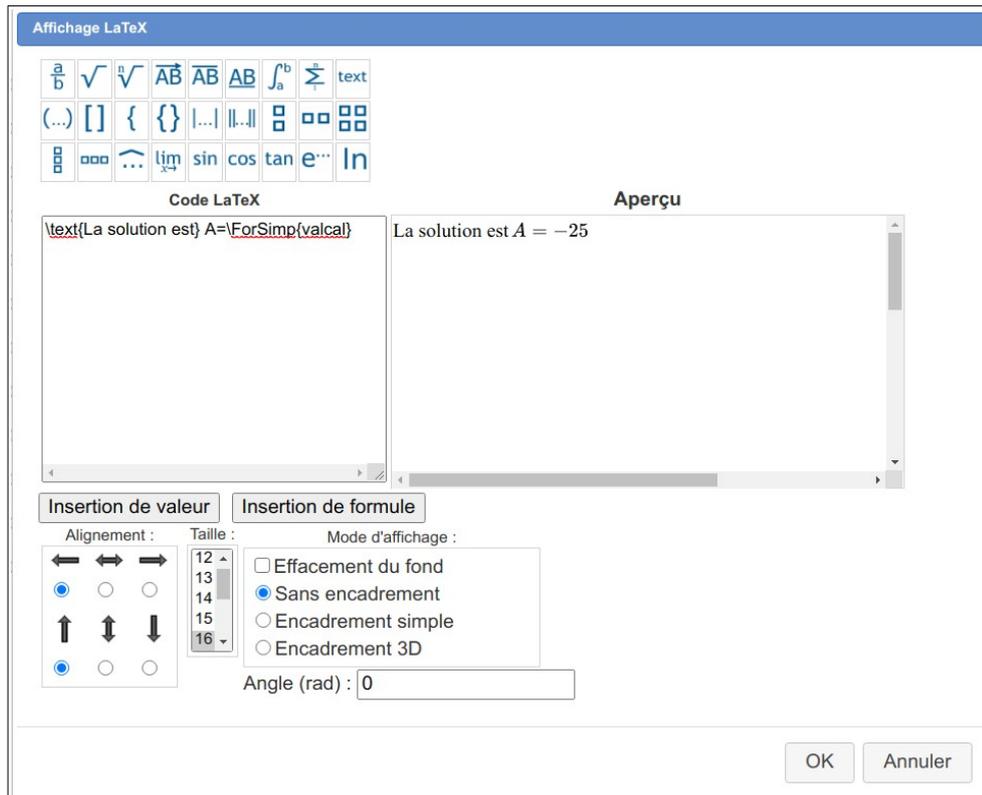


Le code LaTeX `\text{ForSimp}` est un code LaTeX interne à MathGraph32 qui demande ici d'écrire la formule du calcul cal sous la forme la plus simple possible (« ForSimp »). Le code LaTeX de cet affichage sera récupéré par LaboMep pour afficher la consigne de l'exercice (on le mettra en paramétrage, voir à la fin du tuto).

14) Rendez invisible cet affichage (c'est Labomep qui gère l'affichage ce texte, pas

MathGraph32) en le masquant : , puis en cliquant sur le texte affiché.

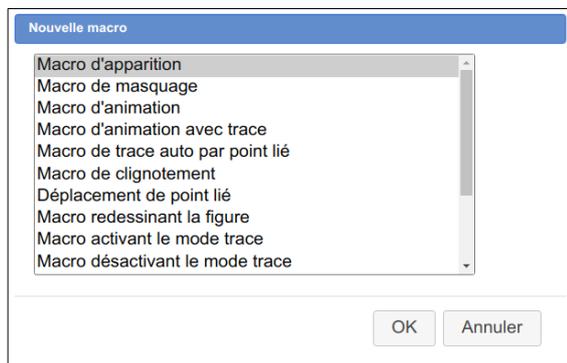
15) L'affichage d'une solution est obligatoire. Utilisez à nouveau  et tapez : « `\text{La solution est} A=\text{ForSimp}{valcal}` », puis validez en cliquant sur « OK » :



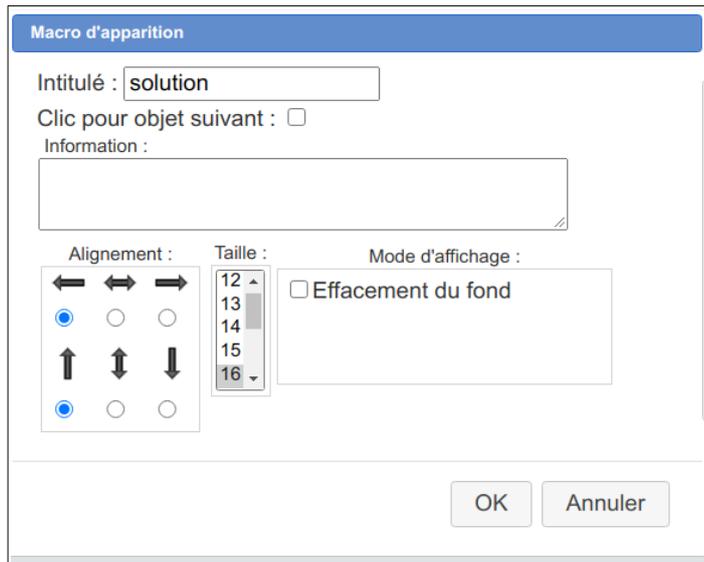
16) Cette solution ne doit pas être visible au départ, **et doit être déclenchée par la fin de la saisie**. Pour cela, il faut créer une 'macro d'apparition' : cliquez sur



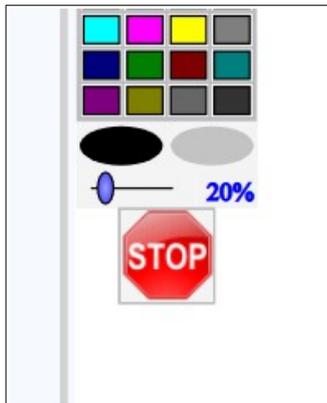
Puis sur 'Macro d'apparition', et validez.



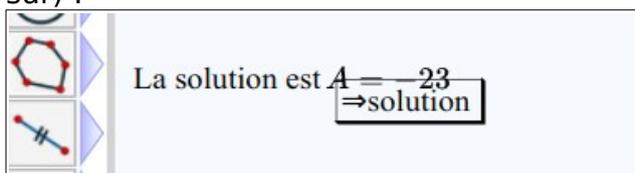
Cliquez sur « La solution est A=... ». Complétez ensuite l'Intitulé avec « solution » et validez :



Cliquez à nouveau sur le texte « La solution est A= ... », puis sur STOP, à droite :

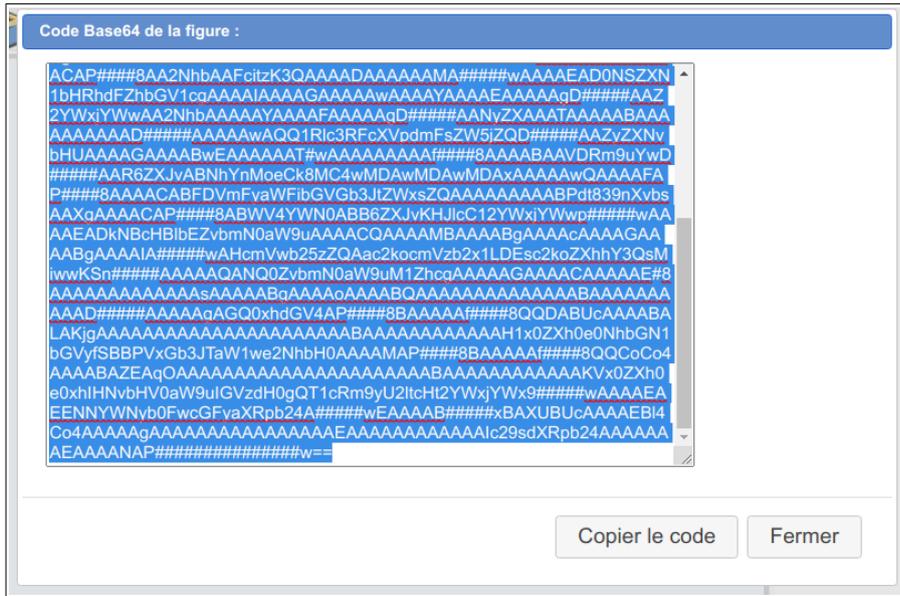


Vous devez obtenir quelque chose de ce genre (la valeur peut être différente, bien sûr) :



17) Masquez la solution **et** le cadre « =>solution » avec .

18) Exportons la figure : , puis cliquez sur « Obtenir le code base64 de la figure » :



Cliquez ensuite sur « Copier le code ». Le code est maintenant dans le presse-papier. Passons à Labomep, pour créer la section :

19) Ouvrez Labomep et connectez-vous.

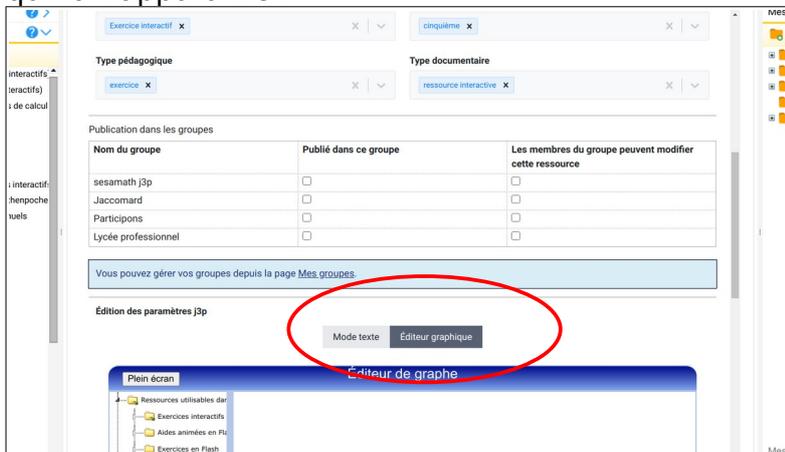
20) Dans 'Mes ressources', cliquez sur 'créez une ressource' » ().

21) Complétez le **Titre** (à votre convenance), choisissez « *activité j3p* » pour le **type technique**, choisissez « *Exercice interactif* » pour les **Catégories**, et choisissez « *cinquième* » pour les **Niveaux**.

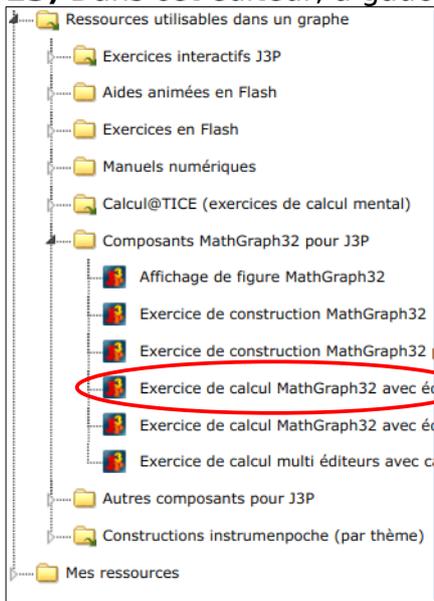
22) Cliquez sur « **Créer la ressource** » : cela va permettre d'afficher l'éditeur de graphe. Celui-ci apparaît tout en bas.



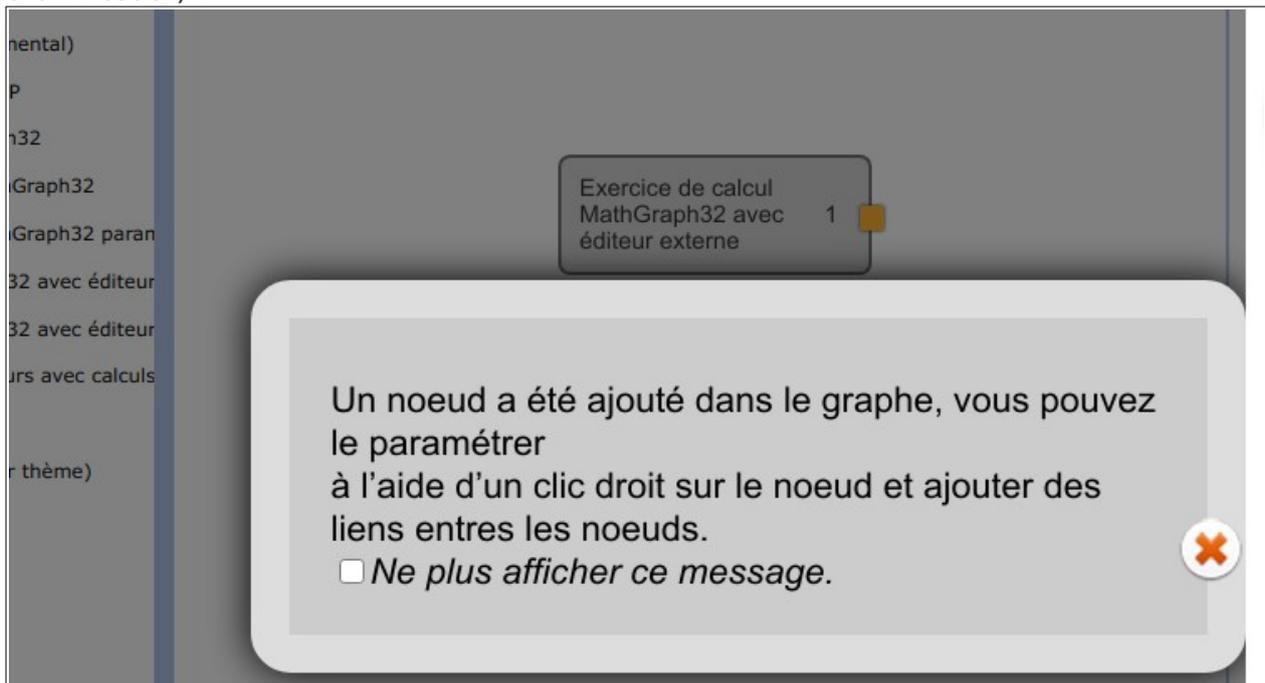
qui fait apparaître :



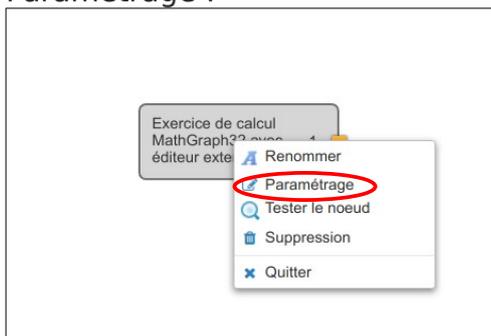
23) Dans cet éditeur, à gauche, déployez 'Composants MathGraph32 pour J3P' :



24) Traînez 'Exercice de calcul MathGraph32 avec éditeur externe' au centre de l'éditeur de graphe et cliquez sur la croix rouge (c'est une indication sur l'utilisation d'un nœud) :



25) Cliquez avec le bouton **droit** sur 'Exercice de calcul... » et sélectionnez sur Paramétrage :



26) La fenêtre de paramétrage s'affiche :

Nous avons plusieurs réglages à y faire (***pour info, vous avez des indications sur sa signification quand vous cliquez dans le champ***) :

27) Dans le champ 'Fig', **coller** le code base64 de la figure et cliquez sur 'Valider'.

28) Dans le champ 'nbrepetitions', vérifiez que c'est à '2'.

29) Dans le champ 'param', entrez '**rst**'.

30) Dans le champ 'NomCalcul', vérifiez qu'il y a '**A**'.

31) Dans le champ 'nbEssais', entrez 3 au lieu de la valeur '**6**' : l'élève pourra donc appuyer 3 fois au maximum sur la touche Entrée pour tester ses calculs intermédiaires avant de cliquer sur OK pour valider sa réponse.

32) Ici le paramètre 'nbchances' ne sera pas utilisé. Il le serait si nous avions mis le paramètre 'validationAuto' à true. Dans ce cas l'élève ferait ses calculs intermédiaires en appuyant sur la touche Entrée, et valide sa réponse finale en cliquant sur le bouton OK. Le paramètre 'nbchances' est alors le nombre maximum de validations autorisées.

33) Dans le champ 'nblatex', laissez la valeur '**1**' : nous n'avons qu'un seul affichage LaTeX de la figure à récupérer pour l'incorporer (le « Calculer A=... »).

34) Mettez le champ 'indicationfaute' à '**false**' : nous n'avons pas prévu d'indication en cas d'erreur (voir une version ultérieure du tuto).

35) Dans le champ 'charset', mettez les caractères autorisés lors de la saisie, ici : ()0123456789+/* (si vous laissez le 'charset' vide, tous les caractères seront autorisés).

36) Dans le champ 'enonceligne1', entrez '\$\LaTeX\$' : \LaTeX signifie que le code LaTeX du premier affichage LaTeX de notre figure sera inséré ici.

37) Les autres 'enonceligne2' à 'enonceligne8', ainsi que 'enoncesimplifier' vides

38) Mettez toutes les cases 'btn...' à '**false**' : elles correspondent à des boutons facilitant la saisie, mais sont inutiles ici, puisqu'on saisit des nombres entiers.

39) Terminez en cliquant sur 'Valider', puis sur 'Enregistrer' (tout en bas) :

16) Cliquez sur l'emplacement voulu sur la figure, puis créez une macro intitulée **solution** (le nom est essentiel). **Attention : ce nom de variable est obligatoire** (il est utilisé par le programme pour afficher la réponse de l'ordinateur) :