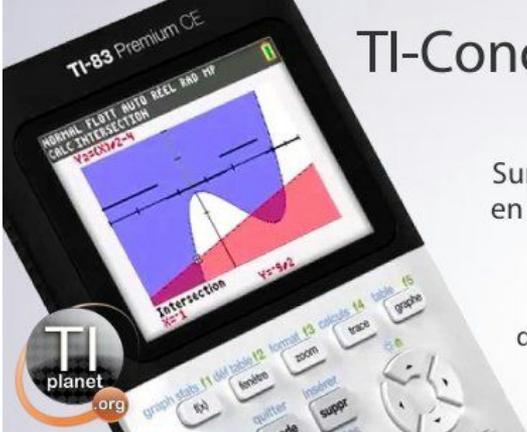


# TI-Concours 2016

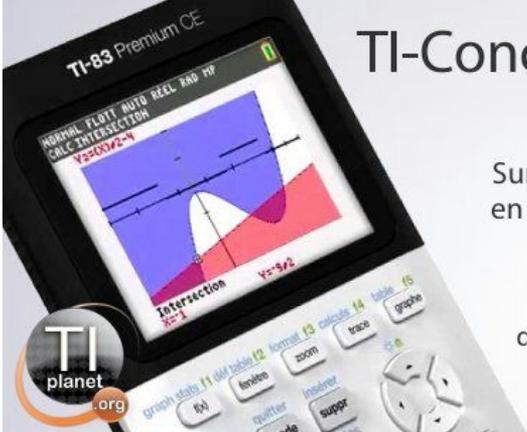
*Qualifications, catégorie 2  
(Lua Nspire et Asm/C eZ80)*

\*\*\*



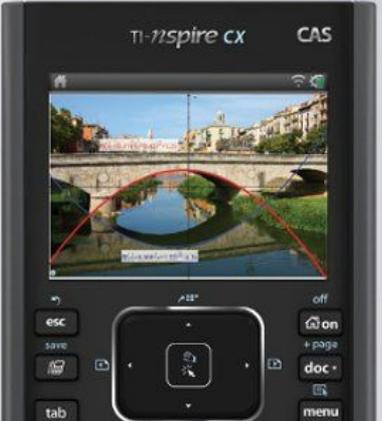
TI-83 Premium CE

NORMAL FLOAT AUTO REEL RAD PP  
CALC INTERSECTION  
Y2=0.5722-4  
Intersection  
X=1  
Y=3.2



TI-84 Plus

graph stats 11 del table 12 format 13 calculate 14 table 15  
linest zoom  
quitter insérer  
mode supp  
test



TI-Nspire CX CAS

esc save tab  
off  
on  
page  
doc  
menu

## TI-Concours 2016

Sur 82/83/84 et Nspire,  
en Basic, ASM, C et Lua

Des calculatrices et  
des goodies à gagner !



## Avant-propos

Ceci est un sujet de qualification, ce qui signifie que vous avez jusqu'au 08/02/2016 23h59 (UTC+1) pour envoyer votre production à l'adresse [info@tiplanet.org](mailto:info@tiplanet.org). Vous mettez la catégorie (Catégorie 2) en objet, et dans le contenu du message vous indiquerez nom, prénom, et adresse postale complète. N'oubliez pas de joindre votre production, qui sera envoyée sous la forme d'une archive ZIP ou RAR, dont le nom sera votre nom de famille en majuscules sans accents, suivi d'un espace et des caractères « 2Q ». Cette archive contiendra l'ensemble des programmes que vous aurez faits (et rien de plus).

Si vous décidez de participer dans les deux catégories, veuillez nous adresser deux courriers électroniques séparés. Vous pouvez mettre à jour votre production autant de fois que vous le souhaitez, en envoyant à chaque fois un courriel comme indiqué ci-dessus. Pour chaque catégorie, nous ne tiendrons compte que de la dernière production reçue dans les délais.

Pour ce sujet, vous devrez pour chaque question écrire un programme, dont le nom sera : les 5 premières lettres de votre nom de famille, toutes attachées et sans accent, suivi des caractères « 2Q » et du numéro de la question. Ainsi, si votre nom de famille est De Périgny, pour la question 4 votre programme aura pour nom « DEPER2Q4 ». En procédant ainsi, vous limitez le risque que deux programmes s'appellent pareil. Si malgré cela vous vous rendez compte que quelqu'un d'autre dont les cinq premières lettres du nom sont les mêmes que les vôtres participe, ne vous en souciez pas, nous nous occuperons du problème.

Les programmes doivent tous être écrits intégralement soit en Nspire-Lua et/ou Nspire-Basic, soit en Asm/C eZ80. Attention, dans ce dernier cas ce sont bien les codes source qu'il faut nous donner, et non les exécutable.

Pensez à vous munir d'un câble permettant la connexion entre votre calculatrice et votre ordinateur, et de vous procurer le logiciel de transfert approprié. Si vous avez des difficultés avec ça, n'hésitez pas à poser des questions sur <https://tiplanet.org>.

Sauf mention explicite du contraire, vous n'aurez jamais à demander une saisie de données à l'utilisateur, ni d'afficher quelque chose en sortie. Le sujet vous donnera des noms de variables, vous indiquera ce que chacune d'elles représente, et vos programmes s'exécuteront sous l'hypothèse que ces variables ont été initialisées correctement au préalable. De même, il vous sera toujours spécifié les variables qui devront contenir le résultat demandé. Vous avez le droit de modifier les variables utilisées en entrée via ce que vous programmez, et on suppose qu'il n'y a jamais d'erreur de saisie.

Le barème précis ne sera pas dévoilé, mais sachez qu'environ la moitié de la note portera sur la justesse des résultats produits, et que la seconde moitié portera sur l'efficacité de l'algorithme implémenté. Si vous finissez longtemps avant la fin, vous pouvez toujours essayer de rendre vos programmes plus efficaces.

La difficulté est globalement croissante, et il n'est pas nécessaire d'avoir tout fait pour vous qualifier pour la suite. Nous vous encourageons à nous soumettre ce que vous avez fait, quel que soit le nombre de questions traitées. Quoiqu'il arrive, nous espérons que vous aurez du plaisir à chercher des réponses !

**Bonne chance à tous !**

# Le sport national

Dans la République Complètement Tarée (ou R.C.T.), les dirigeants ont souvent eu des exigences assez particulières vis à vis de leur peuple. Le gouvernement récemment élu, quand à lui, a une passion immodérée pour le rugby, et a bien l'intention de faire subir son attrait pour le ballon ovale à la population.

## Question 1

On suppose qu'à l'issue d'un match de rugby, l'équipe 1 a obtenu  $A$  points, et que l'équipe 2 a obtenu  $B$  points. Écrire un programme qui récupère les valeurs des variables  $A$  et  $B$  et qui affecte comme valeur à la variable  $C$  1 si l'équipe 1 gagne, 2 si l'équipe 2 gagne, 0 s'il y a match nul.

\*\*\*

Profitant des richesses amassées par l'État au cours du temps, le gouvernement décida, peu après son arrivée au pouvoir, de créer une commission visant à émettre de profondes réflexions sur l'avenir de ce sport s'il venait à devenir ancré dans la tradition nationale.

À la suite de cette décision, ce panel d'experts rédigea un rapport intitulé "Théorie du rugby moderne" dans lequel il propose une généralisation du sport actuellement pratiqué. Selon cette publication, le rugby serait un cas particulier de "sport à plusieurs niveaux de notation", dans lequel plusieurs types de faits de jeu sont récompensés par un barème précis. Ainsi, dans le cas particulier du rugby, on a les niveaux suivants :

- 1) Pénalité : 3 points;
- 2) Drop : 3 points;
- 3) Essai : 5 points;
- 4) Essai transformé : 7 points.

Impressionnés par cette découverte, les membres du gouvernement cherchent à déterminer quel barème rendrait le jeu le plus passionnant. Ils veulent notamment s'assurer qu'il y ait le plus possible de scores atteignables dans une partie.

## Question 2

On suppose que la liste  $L_1$  contient la liste des niveaux dans un ordre quelconque. Ainsi, dans l'exemple précédent on pourrait avoir  $L_1 = \{3, 5, 3, 7\}$ . Écrire un programme qui récupère cette liste  $L_1$  et qui affecte à la variable  $R$  la valeur 1 si tous les scores sont possibles, la valeur 0 sinon.

\*\*\*

Afin que tout le monde puisse profiter de cette avancée majeure, le gouvernement décida de rendre obligatoire la pratique du rugby. À la même occasion il créa une carte d'adhérent permettant à tout le monde d'être pointé aux entraînements, et ainsi s'assurer que personne ne sèche ces derniers. Cette carte devant être retirée à la préfecture par tout le monde, de longues heures d'attente se profilent.

Vous aussi, vous allez devoir aller la chercher !



## Question 5

Réaliser un mini jeu permettant un entraînement spécifique à cette situation. En d'autres termes, le joueur incarnera un personnage se retrouvant à la fin de la queue pour entrer dans un stade, et il devra pouvoir changer de voie quand il le souhaite, le but étant d'entrer le plus rapidement possible dans le stade, sans jamais rentrer en collision avec qui que ce soit. On pourra jouer sur la longueur des queues, ainsi que sur leur nombre. On pourra également indiquer d'une manière ou d'une autre quels spectateurs sont arrivés en groupe. **La jouabilité, ainsi que la qualité des graphismes réalisés seront prises en compte dans la notation.**

\*\*\*

Comme le rugby n'est pas votre sport préféré (la pétanque, c'est moins fatiguant), vous essayez de tuer les matches le plus rapidement possible, c'est à dire après un nombre minimal d'actions. Le problème est que, votre entraîneur définit des objectifs au début de chaque rencontre, et votre équipe est donc obligée d'obtenir un nombre de points supérieur ou égal à cet objectif !

## Question 6

On suppose que la liste  $L_1$  contient la liste des niveaux dans un ordre quelconque. Écrire un programme qui récupère cette liste  $L_1$  et qui stocke dans la variable R le nombre minimal d'actions à réaliser pour atteindre l'objectif initialement fixé, qui est de O points.

\*\*\*

FIN DU SUJET

\*\*\*