

**M1-PE, TP 3. En route vers SCRATCH, en faisant un dernier petit tour du côté du TABLEUR...**

**TP. EXERCICE 1** : des tâches de calculs avec des pourcentages, *trois items indépendants...*

**a.** D'après l'INSEE, la dépense totale en achat de fruits frais des ménages français a augmenté de 7,5% entre 2011 et 2012. Elle était de 9651 millions d'euros en 2011. *Quelle était la dépense totale de ces achats en 2012 ? Arrondir.*

**b.** La dépense totale des ménages français en achat de journaux (*quotidiens, hebdomadaires et mensuels*), se montait à 2542 millions d'euros en 2011. Elle a baissé de 1,28% entre 2011 et 2012. *Calculer la dépense des ménages français en journaux en 2012. Arrondir.*

**c.** L'activité-fret du port de Rouen, en termes de trafics de marchandises, s'élevait à 26,7 millions de tonnes en 2010. Sachant qu'en 2011 l'activité du port de Rouen ne s'élevait plus qu'à 25,4 millions de tonnes. *Calculer le taux d'évolution entre 2010 et 2011. Arrondir.*

*Pas d'indication particulière, ni de corrigé : items de calculs simples (application d'un taux de pourcentage, détermination d'un taux de pourcentage). Ce qui peut être intéressant, en plus, c'est de préciser le choix des arrondis, dans le contexte de chaque item.*

**TP. PROBLEME** : un peu de tout, avec un peu de tableur...



Une entreprise produit et commercialise chaque mois  $x$  milliers de tee-shirts, avec  $0 \leq x \leq 50$ .

On appelle  $C(x)$  le coût total mensuel de production et  $R(x)$  la recette mensuelle réalisée par la vente de  $x$  milliers de tee-shirts. Les coûts  $C(x)$  et  $R(x)$  sont exprimés en milliers d'euros.

*On suppose que toute la production est vendue entièrement chaque mois.*

1. Préparer une feuille de calcul-tableur. On fixera à 5 le pas de l'incrément de la colonne qui donne le nombre  $x$  de tee-shirts. Programmer la fonction coût  $C$  et la fonction recette  $R$ . « Discussion » sur les valeurs affichées...

2. Les deux courbes représentatives, du **coût** (*en trait continu*) et de la **recette** (*en pointillés*), sont données *page suivante*. Répondre alors aux questions suivantes.

Par lecture graphique, déterminer :

- Le coût total de production de 20 000 tee-shirts, la recette réalisée par la vente de ces derniers ainsi que le bénéfice obtenu ;
- Les quantités de tee-shirts à produire et à vendre pour que le bénéfice soit nul.
- Est-il possible de déterminer graphiquement la quantité de tee-shirts à produire et à vendre pour que le bénéfice maximal soit atteint ?

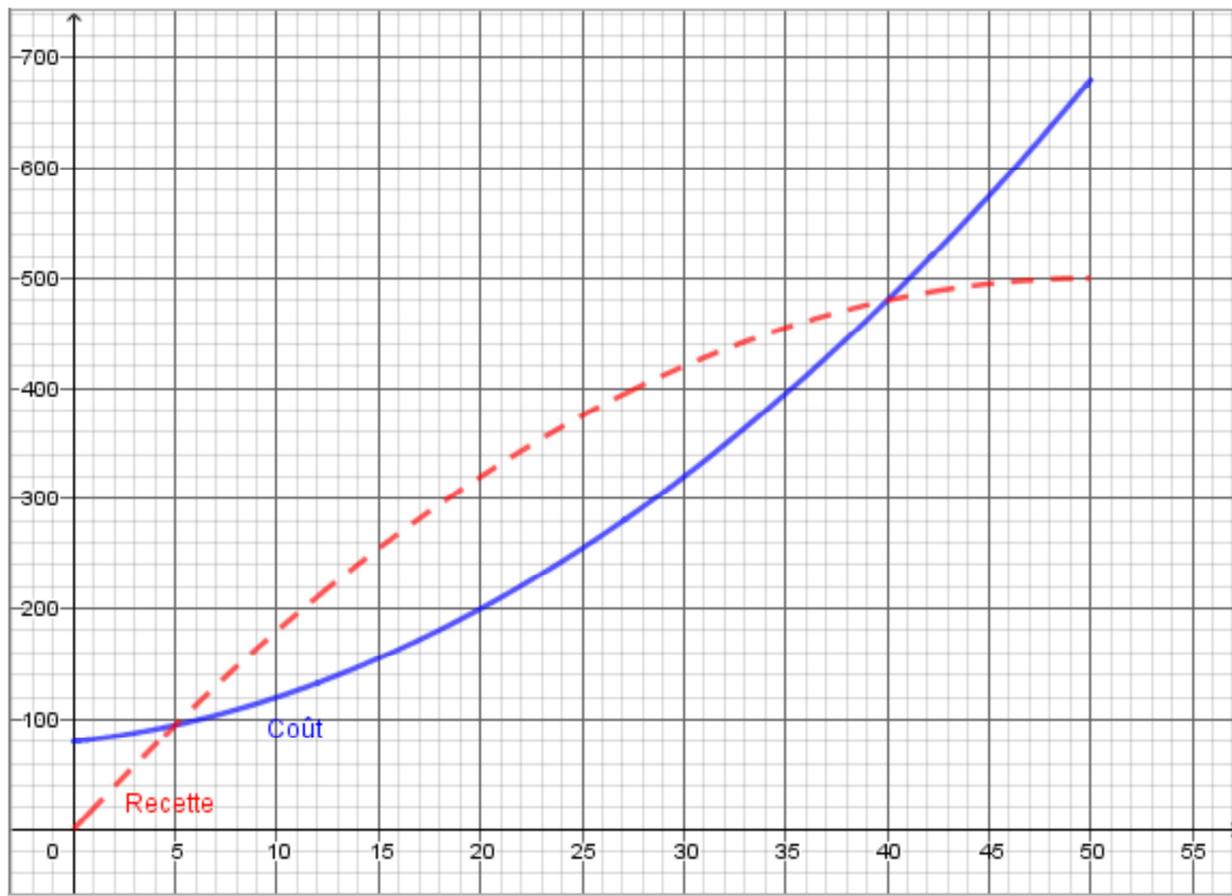
3. On admet que la fonction  $C$  est définie par  $C(x) = 0,2x^2 + 2x + 80$ , entre les valeurs 0 et 50, *toujours en milliers d'unités*.

Le prix de vente unitaire d'un tee-shirt, note  $P(x)$  en euros, varie en fonction du nombre  $x$  de tee-shirts produits et vendus. On admet que  $P(x) = 20 - 0,2x$ .

• Montrer que  $R(x) = -0,2x^2 + 20x$ , puis en déduire que le bénéfice mensuel  $B(x)$ , exprimé en milliers d'euros, réalisé par la production et la vente de  $x$  milliers tee-shirts est défini par  $B(x) = -0,4x^2 + 18x - 80$ .

- Répondre alors algébriquement aux deux questions posées à la question 2.

Graphique à étudier :



Quelques indications :

- Feuille de calcul : une première colonne pour le nombre  $x$  de tee-shirts, avec un pas égal à 5 (ou autre), de 0 à 50. Une deuxième colonne : on programme la fonction *coût*, on n'oublie pas le « = » ! Idem pour la fonction *recette* dans une troisième colonne. Plus éventuellement, une dernière colonne où on peut calculer la différence « *recette - coût* » ou « *coût - recette* ». Si tout est « ok », il y a deux valeurs pour lesquelles la différence est nulle. *Interprétation...*
- Lecture graphique : pas d'indication... Bénéfice maximal : on a un « intervalle de lecture », à justifier géométriquement...
- Calculs algébriques, un peu délicats. Revenir aux sens communs et mathématiques des mots : *coût*, *recette*, *bénéfice*, *prix unitaire*... On se croirait en classe de 1èreES ou mieux, en classe de 1èreSTMG ? *Hihihhi...*

Et maintenant, on est parti avec **SCRATCH** avec le **petit canard jaune** de **PW**, *hihihi*.

PREAMBULE : What's that **SCRATCH** ? *Quelques mots, avant d'aller vite aux faits ...*

Pas de plaidoyer pro-domo pour une étude scolaire de l'*algorithmique* et de la *programmation*, dans le domaine élargi du « *numérique* » : dans le développement de toutes les (*vraies*) sciences aujourd'hui, c'est absolument incontournable. Marche arrière impossible et marche avant obligatoirement réalisée à une **Très Grande Vitesse** !

Voilà : « *Just phone to Simone Driver Application* », *hihihi...* *Commentaires à l'oral...*

C'est ce qui explique cette arrivée en force au cycle III dans les nouveaux programmes.

Cf. les **PROGRAMMES du cycle III** : **OBLIGATOIRE**... Pas de commentaire dans ce document : il FAUT lire et s'appropriier ces programmes !

Le choix institutionnel s'est porté sur le logiciel **SCRATCH**. (C'est à la fois un logiciel de programmation par « blocs », totalement « free », édité par le MIT, mais en même temps un langage de programmation : on fait ainsi d'une pierre deux coups, a minima !).

A notre niveau du Master Meef-PE, on va considérer comme synonymes « un programme de calcul », « un programme de construction géométrique » et un « programme scratch, appelé « script » ». On va simplement dire qu'un *algorithme*, dans un certain « langage », a pour fonction de décrire un programme. Définition du mot « *algorithme* » délicate à notre niveau... Commentaires...

Le but du **TP** est d'étudier quelques exemples emblématiques mettant en jeu des algorithmes simples, mais standards et de les programmer en **SCRATCH**.

Question : « **C'est-y koi ki peut tomber par terre, c'est la faute à Voltaire, au CRPE ?** ». Il faut préciser « le plus important » : oui, mais pour qui ? Le bon « M1-PE » de base !!!

Quelques types de questions qui peuvent (*qui vont ?*) être posées au Concours... Source : manuel de préparation au CRPE 2018, éditions Dunod.

HYPOTHESE 1, hypothèse « *mucho forte* » **PW** : le jour du CRPE, on fait de « *l'informatique débranchée* » (**PW** dixit = on fait de « *l'informatique plate* ») et donc, on doit lire attentivement le programme ou l'algorithme et « faire » comme si on disposait de la dimension inter-active du numérique sur le sujet d'examen, mais la feuille d'examen n'est pas (*encore ?*) « numérisationnable » !, pourtant, on doit faire comme si elle l'était. *Subtil quand même !*

Même hypothèse concernant les feuilles de calcul d'un fichier-tableur ou les programmes de Géométrie Dynamique, avec un « LoGèDy » ou LGD.

Ce qui conditionne donc le type de tâches demandées au **CRPE** : liste précise ci-dessous

- Analyser, coder ou décoder un *programme* ou un *algorithme* qui « fonctionne » (*expliquer pourquoi il « fonctionne », repérer la ou des erreurs éventuelles, corriger, voire mieux, ré-écrire le programme ou l'algorithme, ...*) ;
- Associer un *programme* ou un *algorithme* à une réalisation de celui-ci ;
- Compléter un *programme* ou un *algorithme* pour qu'il « fonctionne », connaissant les données et le « produit » final, ou une autre combinaison ;
- Mettre en forme un *programme* ou un *algorithme*, connaissant une ou des données et le « but » ; ...

HYPOTHESE 2 : « c'est en « **scratchant** » qu'on devient « **scratcheur** ». Le but premier n'est pas de devenir, *tout de suite*, un pro de **SCRATCH**, et encore ; mais surtout de s'intéresser à des problèmes mathématiques où une certaine *approche numérique* et où le *raisonnement algorithmique* sont incontournables. Pas plus, mais pas moins !

HYPOTHESE 3 : pour **scratcher** comme il faut, on doit connaître une catégorisation des problèmes et des spécificités de l'algorithmique (*initiation, initialisation, déroulement, test d'arrêt, calculs, ...*) ; cette catégorisation se fera par une étude d'exemples emblématiques... C'est l'objet du **TP**.

Du côté des (*futurs*) apprentissages : une méthodologie (*simplifiée, non nécessairement linéaire*) de développement d'un programme ou d'un algorithme, basée sur les phases « **e - e - a** », si chères à **PW** !

Rappel : « **e - e - a** » = « **essais - erreurs - ajustements** », très bonne modélisation de ce qu'on met tous derrière le ou les phases de **tâtonnements** dans toute recherche...

<p>(1) <u>Lister les « objets</u> :</p> <p>Déterminer les événements et les actions ;</p> <p>Déclarer les variables ou les listes ;</p> <p>Rédiger les deux scripts : l'algorithme et le programme ;</p> <p>Tester et mettre au point...</p>	<p>(2) <u>Analyser le problème et développer un algorithme « à la main »</u> :</p> <p>Choisir les « structures » informatiques (<i>itération, boucles, tests...</i>)</p>	<p>(3) <u>Rédiger un script-programme</u> :</p> <p>Traduire l'algorithme sous la forme d'une suite d'instructions ; tester...</p>
--	--	---

## Principaux éléments de **SCRATCH**

➤ On commence par les **lutins** (= « objets **SCRATCH** »). **PW** a choisi un **canard jaune**, quelle idée ! Chaque **lutin** possède un **stylo** qui peut donc afficher une trace écrite ou dessinée. Repérage du **lutin** : système d'axes de coordonnées perpendiculaires, avec abscisses comprises entre - 240 et + 240, et ordonnées comprises entre - 180 et + 180 (« unité ponctuelle » = pixel) ;

➤ Les blocs « **aimantés** ». Instructions et programmes appelés « **scripts** » (= nom d'un programme **SCRATCH**) ;

➤ Les blocs « **ovales** » ou « **arrondis** », des blocs « **rectangulaires** » et les blocs « **pointus** »... Chaque forme possède sa spécificité liée aussi à des couleurs.

Les éléments essentiels à TOUTE programmation, indépendamment du logiciel choisi, c'est la partie noble de l'informatique : l'informatique débranchée : une « **instruction** » est représentée par une pièce de puzzle, dans la zone des menus colorés.

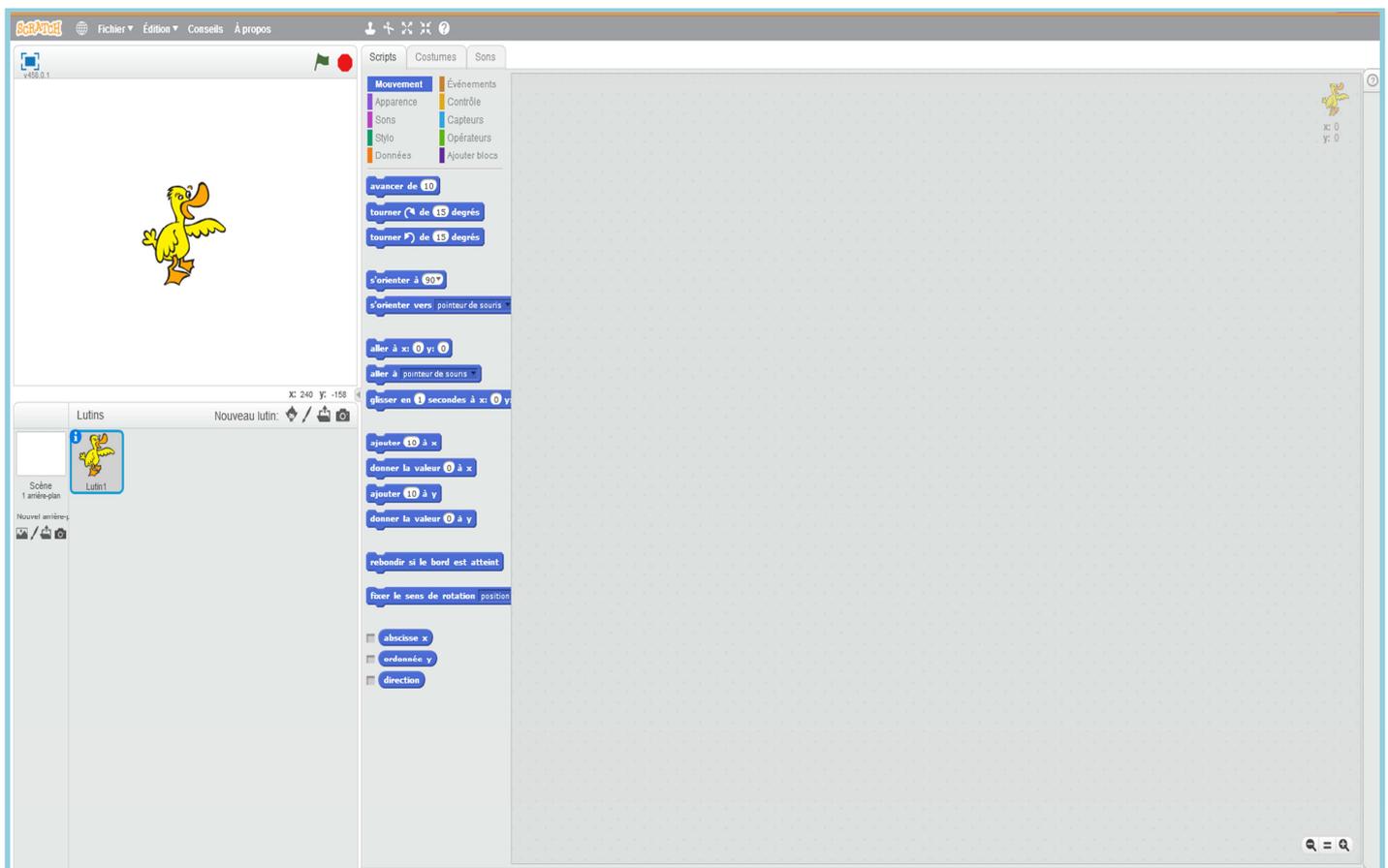
➤ Les « **boucles** », itératives (= *répéter*) ou conditionnelles ((*si... alors*) ou (*si... alors... sinon*) ou (*répéter jusqu'à...*) ou (*répéter ... fois*)) ;

➤ Les « **tests** » ; les « **opérateurs** » ; les « **blocs s'instructions** » = spécificité **SCRATCH** ;

➤ La notion de **variable** (*informatique*) et la notion de **liste** (*informatique*). *Pas si simple !*

On y est presque : il faut encore voir à quoi ressemble un « écran-**SCRATCH** »

Capture d'écran de l'écran-**SCRATCH**, commentaires à l'oral : *description de chaque zone...*



Ouvrir un « écran-**SCRATCH** ». Choisir un « **lutin** ».

Choisir (éventuellement) un « **arrière-plan** ». On va s'amuser : et oui, c'est pour le **CRPE** !!!

**EXERCICE 1 : premier programme emblématique ! Constructions géométriques...**

- Tracer un carré

Un premier programme	Qui donne ceci

« Analyser » le script ci-contre.

Faire fonctionner le programme plusieurs à plusieurs « endroits » de la scène : (Instruction bleue : mouvement, « coordonnées »).

Pour aller plus loin. Ne peut-on pas « améliorer » ce script ? (Instruction jaune, contrôle : « répéter ... fois »).

Faire « disparaître » le lutin. Autre construction : passer par les coordonnées, subtil ! Cf. les pistes de correction...

**EXERCICE 1.1. (i)** Programmer deux **scripts-Scratch** (*idem ci-dessus*) permettant de construire un losange de côté 120 et d'un angle de 50°.

(ii) Reproduire le **script-Scratch** ci-contre et préciser la nature de la figure géométrique obtenue.

**EXERCICE 1.2.** Programmer un **script-Scratch** qui réalise la construction ci-contre : trois carrés « décalés vers la droite » (*même longueur de côté avec épaisseurs et couleurs distinctes*).



**EXERCICE 1.3.** Tester chaque **script-Scratch**, en vrai, avec le logiciel ; répondre aux questions, puis « améliorer »...

<p>Quelle figure obtient-on avec ce programme ?</p> <p>.....</p> <p>.....</p>		<p>Tracer à main levée la figure obtenue avec ce programme.</p>	
---	--	---	--

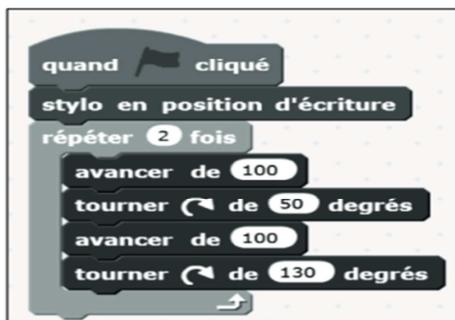
**EXERCICE 2** : d'après CRPE 2017, source : **Annales COPIRELEM...**

Il manque certaines **couleurs**, pourtant indispensables, sorry !!!

**PROGRAMME A**



**PROGRAMME B**



Consigne : Déterminer, ~~sans~~ justifier, quelle figure géométrique est tracée lorsqu'on exécute chacun des deux programmes ci-contre. Copie sans couleur, zut !

Justifier ! Compléments à l'oral...

Réponses...

Autres programmes **Scratch** : du côté du **numérique**... Copie sans couleur, encore zut !

**EXERCICE 2 :**

On utilise le programme ci-contre.

1. Quel résultat s'affiche si l'on choisit d'entrer le nombre 7 ?
2. Quel résultat s'affiche si l'on choisit d'entrer le nombre 12,7 ?
3. Quel résultat s'affiche si l'on choisit d'entrer le nombre -6 ?



Quelques « aides » pour « désosser » (nature des blocs ou des briques ?) le script ci-contre, sans oublier de le programmer, of course.

1) On a créé une VARIABLE (fond de couleur : orange) ; on va donc entamer un « dialogue » avec cette variable. Le résultat de ce dialogue sera « visible » dans la partie où se trouve le lutin ;

2) Il y a une condition : *SI (...)* *ALORS (...)* *SINON (...)* : à explorer ;

3) Le script se termine par une « phrase » qui contient la valeur calculée de la variable... **Script-Scratch** ci-dessus à reproduire, avec les couleurs et à analyser.

Pour aller plus loin. En suivant le script ci-dessus, réaliser un script du même modèle qui renvoie le calcul :  $x^2 - 2x + 2$ , en appelant  $x$  la variable, à l'instruction « *SI - ALORS* » et  $-3x + 13$  à l'instruction « *SINON* ».

On va plus encore plus loin : deux problèmes emblématiques. Consigne : rédiger un script-**SCRATCH** pour répondre à chacun de ces deux problèmes.

**Pb1.** Un nombre entier naturel donné est-il dans une table de multiplication choisie ?

**Pb2.** On se donne plusieurs nombres, les ranger (par ordre choisi). Commencer par chercher à ranger trois nombres (entiers ou autres ou « mélangés »).

Conseil PW : proposer tout d'abord une analyse dite « analyse débranchée » du problème, afin d'écrire un algorithme à la main, puis de produire un **script-Scratch**.

EXERCICE 3 : Construction de quadrilatères particuliers

Analyse mathématique et analyse « débranchée » : de combien d'informations numériques et géométriques a-t-on, besoin pour tracer un quadrilatère particulier ? Parmi ces quadrilatères particuliers, quel est le « moins » particulier ?

Nature et nombre d'informations à demander ?

CONSIGNE : Ecrire un algorithme à la main qui répond aux questions ci-dessus, puis produire un *script-Scratch*. Tester, améliorer, ...

EXERCICE 4 : un algorithme emblématique, **la dichotomie**

Enoncé du problème. On se donne deux nombres  $x$  et  $y$ . Un nombre  $w$ , compris entre  $x$  et  $y$  est choisi. Trouver  $w$ , en proposant des nombres où la réponse produite est « plus grand » ou « plus petit ».

Faire une analyse mathématique et « débranchée du problème ». Produire ensuite un *script-Scratch*.

EXERCICE 5 : se reporter aux **annales COPIRELEM...**

*Deuxième partie de la brochure* : sujets des ESPE, à partir de la page 139...

« Balayer » les exercices proposés et les résoudre ! C'est simple !

Hors contexte **Scratch**, *rappel PW*.

Tout comme il est important de bien chercher à résoudre les exercices-**Scratch**, il est tout aussi important de chercher à résoudre les exercices se rapportant au troisième problème du sujet de CRPE. Voilà !

**PISTES de CORRECTION**

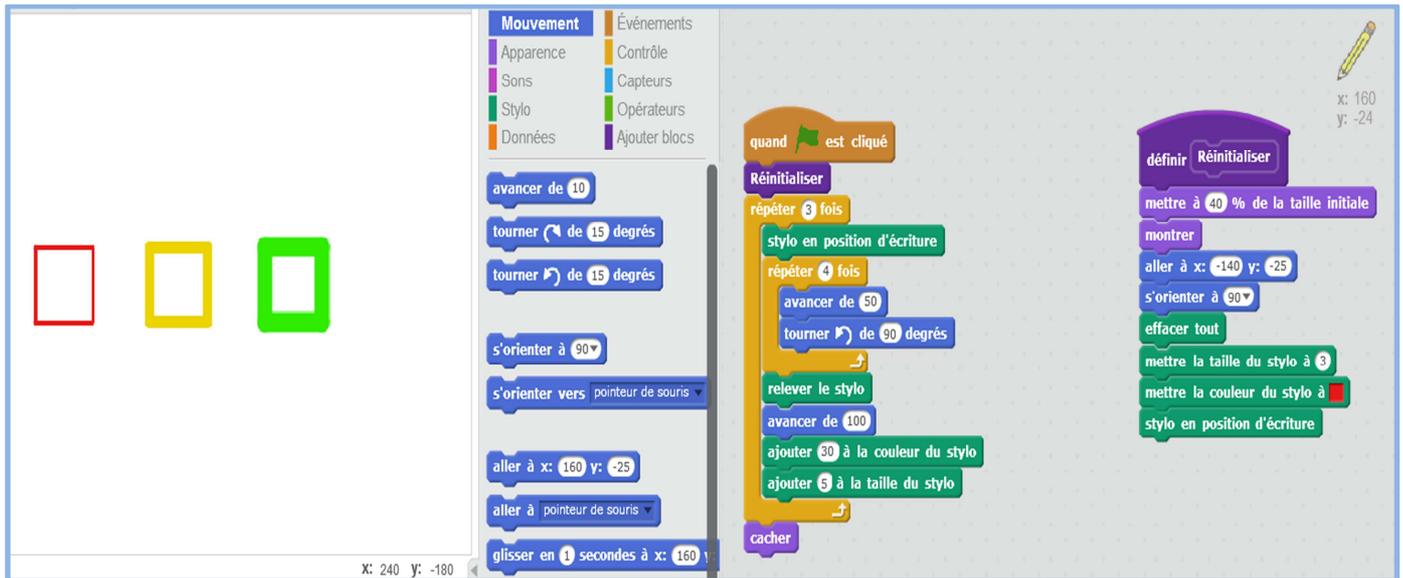
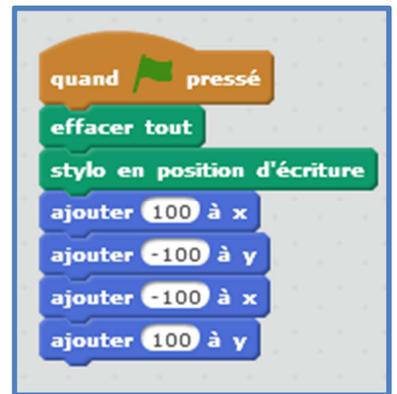
EXERCICE 1 : un carré avec les coordonnées, *subtil*, bis !

EXERCICE 1.1 : pas de correction. Revoir les propriétés d'un losange et les appliquer pour réaliser le *script-SCRATCH*.

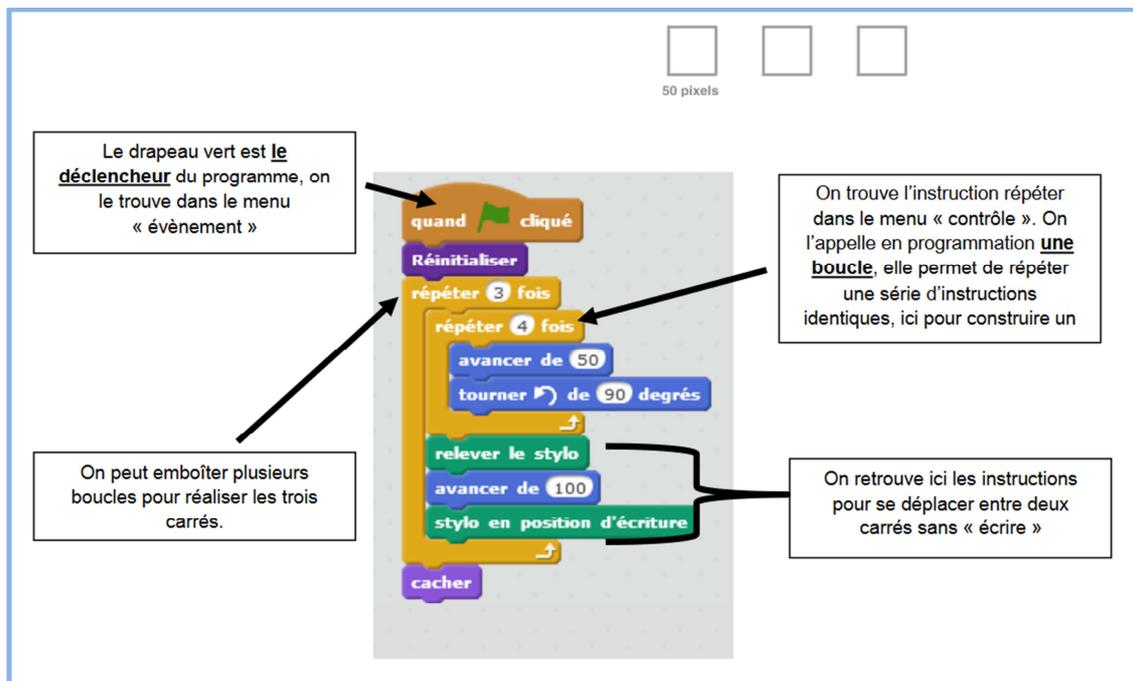
EXERCICE 1.2 : ça devient intéressant.

*Objectif* : parcourir l'écran central des *mini-scripts* (*mouvement, stylo, évènements, contrôle et ajouter blocs*).

Il y a un élément nouveau : on fabrique ici un « bloc », avec le *mini-script* « Réinitialiser ». Cf. ci-dessous...



Des explications ci-dessous : à étudier attentivement...



EXERCICE 2 : les deux problèmes

**Problème 1.**

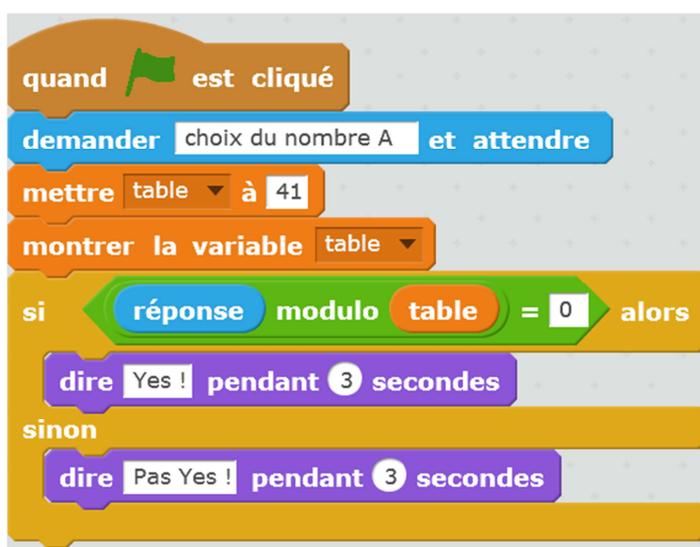
1) Analyse mathématique. « Le nombre entier **A** est dans la table de multiplication de l'entier **n** » signifie : (énoncés synonymes)

- « **A** est un multiple de **n** » ou « **n** est un diviseur de **A** ». *Revoir les définitions...*
- Le reste de la division de **A** par **n** est égal à 0.

2) Analyse dite « débranchée ». Il y a donc une opération arithmétique, laquelle ? Ensuite, existe-t-il un mini-script dans le menu des mini-scripts qui s'intéresse à cette opération ? Enfin, est-ce que tous les restes valent 0 ? On doit donc produire un énoncé conditionnel de la forme *SI (...) ALORS (...) SINON (...)*.

3) Quels scripts ? (i) Demander la valeur de **A** et indiquer celle de **n**. (ii) Produire la condition. (iii) Rédiger proprement le script-**Scratch**.

4) Une solution particulière : nombre divisible par 41 ou non divisible par 41 ?  
Ne pas hésiter à améliorer le script-**Scratch** ci-dessous !



**Problème 2.**

Même type d'analyse que pour le problème 1. A rédiger et à programmer ensuite.

1) et 2) Analyse mathématique et analyse « débranchée ». Pour ranger trois nombres par ordre croissant, ce n'est pas trop long à rédiger à la main.

On se donne donc trois nombres entiers (*non tous égaux, sinon, c'est moins rigolo*) **p**, **s** et **g**. Ranger ces trois nombres, c'est les comparer « deux à deux », puis conclure.

Ce qui donne un programme-algorithme « à la main » décrit ci-dessous : il y a sûrement mieux ! Par exemple, on peut aussi réaliser un arbre » (*il n'y a pas trop de branches, ni trop de feuilles...*).

- (i) SI **p < s** (Ce qui serait sympa, ce serait de comparer **s** et **g**, avec le même « verbe ») ;
- (ii) SI **s < g** ALORS (Afficher **p < s < g**)
- (iii) SINON
- (iv) SI (**p < g**) ALORS (Afficher **p < g < s**)
- (v) SINON (Afficher **g < p < s**)
- (vi) Fin de SI

Traduire ensuite le programme-algorithme en script-**Scratch** : au travail ! (*Non scanné dans ce corrigé*).

EXERCICE 3.

Quadrilatère particulier le « moins » particulier : le parallélogramme. Pourquoi ?

Informations minimales : une « longueur », une « largeur » et un « angle », d'où le *script-Scratch* ci-dessous :

```

quand le drapeau vert est cliqué
effacer tout
stylo en position d'écriture
demander "Quelle est la longueur ?" et attendre
mettre longueur à réponse
demander "Quelle est la largeur ?" et attendre
mettre largeur à réponse
demander "Donner la mesure d'un angle" et attendre
mettre angle à réponse
répéter 2 fois
  avancer de longueur
  tourner de angle degrés
  avancer de largeur
  tourner de 180 - angle degrés
relever le stylo
  
```

EXERCICE 4. Un exemple.

```

quand ce lutin est cliqué
mettre a_trouver à nombre aléatoire entre -50 et 50
mettre reponse à 100
mettre score à 0
répéter jusqu'à a_trouver = reponse
  demander "tape un nombre entre -50 et 50" et attendre
  mettre reponse à réponse
  si reponse < -50 ou reponse > 50 alors
    dire "Attention, il faut taper un nombre entre -50 et 50 !" pendant 2 secondes
  sinon
    si reponse < a_trouver alors
      dire "Plus grand !" pendant 2 secondes
    si reponse > a_trouver alors
      dire "Plus petit" pendant 2 secondes
  ajouter à score 1
dire "regroupe Bravo, tu a fait un score de score" pendant 2 secondes
  
```