



# Exercices Scratch 3ème corrigés PDF : réviser vite pour le Brevet

Téléchargez les bons exercices Scratch 3ème corrigés PDF : variables, boucles, conditions et méthodes efficaces pour le Brevet.

Cours de mathématiques niveau

Mis à jour le 24 avril 2026

**Les exercices Scratch 3ème corrigés en PDF servent à réviser l'algorithmique du Brevet avec des sujets sur variables, boucles, conditions et déplacements. Le meilleur support propose des exercices progressifs, une correction expliquée et des questions proches des annales de maths.**

Votre enfant bloque sur une boucle « répéter » ou confond encore variable et condition ? C'est souvent à ce moment-là que la recherche d'un bon PDF devient urgente. En 3ème, Scratch n'est pas un simple logiciel ludique : c'est un vrai support pour comprendre l'algorithmique, réussir certains exercices de maths et gagner des points au Brevet. Comme parent, élève ou enseignant, le plus utile n'est pas seulement de trouver des fiches imprimables, mais de choisir des exercices progressifs, avec correction claire, niveau adapté et erreurs fréquentes bien expliquées.

## En bref : les réponses rapides

**Quels types d'exercices Scratch tombent le plus souvent au Brevet ?** — Les sujets portent souvent sur la lecture d'un script, les programmes de calcul, les boucles, les conditions et parfois des tracés géométriques ou des déplacements sur repère.

**Un élève de 4ème peut-il utiliser ces exercices Scratch de 3ème ?** — Oui, surtout les exercices de base sur les variables, boucles et déplacements, mais les exercices type Brevet demandent davantage d'autonomie et d'analyse.

**Faut-il savoir programmer entièrement sur ordinateur pour réussir au Brevet ?** — Non, il faut surtout savoir lire, comprendre, compléter et corriger un script Scratch, même sans créer un projet complet depuis zéro.

**Comment imprimer une bonne fiche de révision Scratch ?** — La fiche utile tient sur une ou deux pages, regroupe les blocs essentiels, un exemple de programme de calcul et les erreurs fréquentes à vérifier avant de rendre sa copie.

## Exercices Scratch 3ème corrigés PDF : ce qu'il faut vraiment réviser pour le Brevet

Pour réussir des **exercices Scratch 3ème corrigés PDF**, il faut maîtriser six bases : **variables**, boucles, conditions, déplacements, programmes de calcul et lecture d'un script. Pour le **Brevet**, un bon support ne se limite pas à des fiches à imprimer : il propose des exercices progressifs, une **correction** expliquée et des questions proches des annales de maths.

En **3ème**, l'**algorithmique et programmation** évaluée au Brevet de **mathématiques** reste très concrète. Avec **Scratch**, l'élève doit comprendre ce que fait un script, prévoir un résultat, compléter des blocs manquants ou corriger une erreur logique. Les sujets classiques demandent de piloter un lutin, d'utiliser des coordonnées dans un repère, de tracer une figure, de répéter une action avec une boucle, ou de tester une condition du type  $x < 5$ . On retrouve aussi le **programme de calcul Scratch** : choisir un nombre, lui appliquer des opérations, stocker le résultat dans une variable, puis comparer l'expression obtenue à une écriture littérale comme  $2x + 3$  ou  $x^2 - 4$ . Les élèves de **4ème** ont souvent déjà vu les bases, mais en 3ème on attend plus d'autonomie, plus de lecture de script et une vraie capacité à relier *programmation* et raisonnement mathématique.

La bonne révision ne consiste donc pas à accumuler des **exercices scratch pdf** trouvés au hasard. Un vrai **scratch brevet 3ème** utile doit ressembler aux annales, varier les niveaux et expliquer pourquoi une réponse est juste ou fautive. C'est exactement l'enjeu d'un **exercice scratch brevet corrigé pdf** fiable : aider l'élève à identifier le bon bloc, le bon test conditionnel, la bonne variable, ou la boucle qui manque. Ici, la promesse est plus utile qu'un simple téléchargement de **PDF** : apprendre à choisir un support qui fait progresser, puis apprendre à repérer ses erreurs typiques avant le jour J. Autrement dit, ces **exercices scratch 3ème corrigés pdf** servent à réviser vite, mais aussi à mieux comprendre ce que le Brevet attend vraiment en **algorithmique**, en **programmation** et en maths.

## Comment reconnaître un bon PDF d'exercices Scratch corrigés en 3ème

Un bon **pdf scratch 3ème** se repère vite : **progression réelle**, consignes nettes, corrigés expliqués et situations variées proches du **Brevet des collèges**. Si le document montre seulement le programme final sans commenter les blocs *quand répéter*, *quand tester*, *quand initialiser*, il sert peu pour réviser efficacement et encore moins pour corriger ses automatismes.

Je conseille de juger un **PDF** avant téléchargement avec une grille simple. La **lisibilité** compte d'abord : police claire, une consigne par tâche, espace pour écrire, captures d'écran utiles ou pseudo-code quand le script devient long. Ensuite, regardez le **niveau progressif** : un bon support commence par variables, conditions et boucles, puis combine plusieurs blocs dans un même problème. Beaucoup de ressources en ligne mélangent tout : compilations d'**exercice scratch corrigé** sans ordre, **fiches de révision scratch** trop théoriques, **cours de scratch pdf** complets mais pauvres en entraînement, ou activités prévues dès la 4ème qui ne collent pas aux attentes de 3ème. Le bon document annonce aussi un **temps estimé** par exercice. C'est un détail décisif pour réviser en séances courtes de 15 à 25 minutes, sans épuiser l'élève ni sous-estimer la difficulté.

Critère	Ce qu'il faut voir	Score
<b>Lisibilité</b>	Consignes courtes, mise en page aérée, place pour répondre	0 à 2
<b>Progression</b>	Facile → moyen → type Brevet, sans saut brutal	0 à 2
<b>Aides visuelles</b>	Captures Scratch, blocs visibles, pseudo-code ou schéma	0 à 2
<b>Corrigés</b>	Explication des choix, erreurs possibles, pas seulement la réponse	0 à 2
<b>Proximité Brevet</b>	Situations proches des sujets, logique d' <b>annales</b> , vocabulaire scolaire	0 à 2
<b>Usage pratique</b>	Imprimable, noir et blanc lisible, durée indiquée	0 à 2

Faites le total sur 12. En dessous de 6, le support est souvent décoratif. Entre 7 et 9, il peut compléter un cahier d'exercices. À partir de 10, vous tenez une vraie base de révision. Les meilleures ressources



croisent l'esprit des **annales brevet scratch** avec des corrigés commentés, pas avec des scripts copiés-collés. Mon conseil pratique : constituez un petit dossier de révision en trois pièces, un **cours** synthétique, une **fiche de révision** d'erreurs fréquentes, puis un lot court de PDF imprimables classés par difficulté. Ce trio évite de collectionner des documents redondants et donne un cap clair, séance après séance.



*Brevet Scratch exercices corrigés 2021 — Quarante Douze*

## Mini diagnostic imprimable : ton niveau avant de commencer

Avant d'ouvrir un PDF d'exercices Scratch 3ème corrigés, fais ce **mini diagnostic** en 2 minutes. Réponds sans aide à 5 questions : si un bloc **répéter 4 fois** contient "avancer de 10", le sprite parcourt-il 10 pas ? Une **variable** stocke-t-elle une seule valeur, alors qu'une liste en garde plusieurs ? Dans un programme de calcul qui part de 3, ajoute 5 puis multiplie par 2, obtiens-tu 16 ? Si un test "si score > 10" ne marche pas pour 10, faut-il écrire score > 10 ? Enfin, si le sprite devait aller à droite mais monte, quel bloc de **déplacement** vérifies-tu d'abord : avancer, tourner ou aller à 0 ? Compte 1 point par bonne réponse. 0 à 2 : profil débutant, commence par les exercices guidés. 3 **ou** 4 : profil intermédiaire, vise les exercices progressifs. 5 **sur** 5 : tu es prêt pour un exercice type *Brevet* avec corrigé commenté.

## 4 exercices Scratch originaux et corrigés, classés du plus simple au niveau Brevet

Pour progresser vite, mieux vaut faire **peu d'exercices** mais bien choisis. Les quatre exercices ci-dessous couvrent l'essentiel en **Scratch** en 3ème : déplacement, **programme de calcul**, **condition**, puis sujet complet type **Brevet** avec correction commentée bloc par bloc. C'est le format le plus utile pour des *exercices scratch 3ème corrigés pdf* vraiment exploitables.

En 3ème, Scratch sert à traduire une idée en **algorithme**. On manipule surtout une **boucle**, une **variable** et une **condition**. Au Brevet, le correcteur attend un raisonnement logique, pas seulement un écran qui "marche".

## Exercice 1

**Énoncé :** Sur un quadrillage, le lutin doit avancer de 4 cases vers la droite, puis monter de 3 cases, en utilisant au moins une **boucle**. **Compétence visée :** répéter une instruction de déplacement. **Erreur fréquente :** mettre “avancer de 4” une seule fois au lieu de répéter un déplacement d’une case. **Ce qu’un correcteur attend :** une suite claire d’actions, avec répétition justifiée.

### Voir le corrigé

On programme d’abord une boucle “répéter 4 fois” avec “changer x de  $4$ ”. Puis une boucle “répéter 3 fois” avec “changer y de  $3$ ”. Le choix de  $4$  correspond à une case. C’est simple, mais très formateur. L’élève comprend qu’une boucle remplace des blocs répétés. En géométrie repérée, cela revient à passer d’un point à un autre par translation horizontale puis verticale.

## Exercice 2

**Énoncé :** Demander un nombre, le stocker dans une **variable**  $x$ , calculer  $2x + 5$ , puis afficher le résultat. **Compétence visée :** construire un *programme de calcul scratch*. **Erreur fréquente :** oublier d’utiliser la réponse dans la variable. **Ce qu’un correcteur attend :** saisie, calcul, affichage, dans cet ordre.

### Voir le corrigé

On utilise “demander un nombre”, puis “mettre  $x$  à réponse”. Ensuite, “mettre résultat à  $2 \times x + 5$ ”, puis “dire résultat”. La correction commentée doit faire le lien avec les expressions littérales : Scratch exécute exactement l’expression  $2x + 5$ . C’est un bon *algorithme 3ème pdf* à imprimer, car il relie calcul littéral et programmation.

## Exercice 3

**Énoncé :** Demander un nombre  $n$ . Si  $n$  est pair, afficher “pair”. Sinon, si  $n > 10$ , afficher “impair et supérieur à 10”. Sinon, afficher “impair et petit”. **Compétence visée :** tester une **condition**. **Erreur fréquente :** confondre “pair” et “supérieur à 10”. **Ce qu’un correcteur attend :** une hiérarchie logique correcte des tests.

## Voir le corrigé

On teste d'abord si le reste de la division de  $n$  par  $2$  vaut  $0$ . En Scratch, cela passe par le bloc " $n \bmod 2 = 0$ ". Si oui, le nombre est pair. Sinon, on teste si  $n \bmod 10$ . Cette étape est classique en *exercice scratch technologie 3ème corrigé* : l'élève doit comprendre qu'une condition peut en contenir une autre. On corrige donc la logique, pas seulement le résultat.

### Exercice 4

**Énoncé** : Le lutin trace un carré de côté  $40$ . Après chaque carré, il augmente la longueur de  $10$ . Si la longueur dépasse  $60$ , il affiche "stop". **Compétence visée** : combiner **géométrie**, variable, boucle et condition dans un *exercice scratch brevet corrigé*. **Erreur fréquente** : oublier d'actualiser la variable côté. **Ce qu'un correcteur attend** : un enchaînement cohérent et commenté.

## Voir le corrigé

On crée la variable "côté", initialisée à  $40$ . Puis on répète : tracer un carré avec une boucle de 4 côtés, chaque côté étant "avancer de côté" puis "tourner de  $90^\circ$ ". À la fin, on ajoute  $10$  à "côté". Ensuite, on teste si "côté  $> 60$ ". Si oui, on dit "stop". Cet exercice fait sentir la progression vers le Brevet : la **boucle** pilote la figure, la **variable** fait évoluer la taille, la **condition** contrôle l'arrêt. C'est exactement l'esprit des meilleurs **exercices scratch 3ème corrigés pdf**.

## Corrigés commentés : ce qu'il faut expliquer à l'oral ou à l'écrit

Un bon corrigé ne donne pas seulement la solution : il montre le **raisonnement**. Pour un exercice Scratch 3ème corrigé PDF, l'élève doit dire *ce que fait le script*, pourquoi les blocs sont placés dans cet ordre, puis vérifier si le résultat obtenu correspond à l'objectif demandé. C'est cette logique qui compte au **Brevet**, à l'oral comme à l'écrit.

La correction devient claire si l'on suit quatre réflexes. D'abord, identifier l'**objectif du script** : déplacer un lutin, calculer un score, répéter une action. Ensuite, nommer le rôle de chaque bloc : *la boucle répète l'action, la variable mémorise la valeur, le test décide selon la condition*. Puis, justifier l'ordre des instructions : initialiser avant de calculer, tester avant d'agir, répéter jusqu'au résultat attendu. Enfin, vérifier concrètement la sortie du programme : valeur affichée, position du lutin, message final. Dans un exercice Scratch 3ème corrigé PDF, cette mise en mots transforme une correction brute en réponse compréhensible, réutilisable et plus solide le jour de l'épreuve.

## Méthode anti-erreurs : quel bloc Scratch vérifier quand un exercice ne marche pas

Quand un **script Scratch** échoue, vérifie toujours dans le même ordre : **événement de départ**, **variables**, **condition**, **boucle**, puis **coordonnées** ou affichage. Cette routine simple évite de tout modifier au hasard. Pour les *erreurs scratch brevet*, elle aide à corriger vite, aussi bien en maths qu'en *exercice scratch 3eme technologie*.

Je conseille une lecture en chaîne, presque mécanique. Si le personnage ne bouge pas, le **bloc Scratch à vérifier** est d'abord l'événement : *quand drapeau vert est cliqué*, *quand cette touche est pressée*, ou un message jamais reçu. Ensuite seulement, regarde les blocs de mouvement, car un déplacement correct ne sert à rien si le script ne démarre jamais. Si le résultat affiché est faux, cible les **variables** : valeur initiale oubliée, variable jamais mise à jour, ou calcul mal ordonné, par exemple si l'on veut  $2 \times (3 + 4)$  mais que le script calcule  $(2 \times 3) + 4$ . Si le script tourne sans fin, la **boucle** est presque toujours responsable : *répéter indéfiniment* placé au mauvais endroit, compteur qui n'évolue pas, ou condition d'arrêt impossible à atteindre. Pour **corriger un script Scratch**, il faut donc remonter à la cause logique, pas seulement au symptôme visible.

Autre réflexe utile en **révision scratch 3ème** : relier chaque panne à une famille de blocs. Un test qui ne se déclenche jamais renvoie à la **condition** : signe inversé, variable comparée au mauvais nombre, texte comparé à un nombre, ou capteur jamais actualisé. Une figure décalée, un lutin qui dessine au mauvais endroit, ou un tracé asymétrique pointent plutôt vers les **coordonnées**, l'orientation et parfois le stylo :  $\pi$  et  $\pi$  inversés, angle de départ faux, ou retour à la position initiale oublié. En technologie comme en maths, cette grille fait gagner du temps, car elle transforme une impression de bug flou en diagnostic précis. Voici la version mentale la plus efficace : **ne bouge pas** → événement + mouvement ; **résultat faux** → variables + calcul ; **tourne sans fin** → boucle ; **test muet** → condition ; **figure décalée** → coordonnées + orientation.

Pour relire sans paniquer, applique une méthode en **4 étapes**. D'abord, lis le script comme une histoire courte : que doit-il faire, et dans quel ordre ? Ensuite, repère un seul bug visible, pas dix à la fois. Puis vérifie les blocs dans l'ordre anti-erreurs : événement, variables, condition, boucle, coordonnées. Enfin, teste après chaque petite correction, car modifier plusieurs blocs d'un coup masque souvent la vraie cause. Cette discipline paraît simple, néanmoins elle distingue souvent l'élève qui tâtonne de celui qui sait **corriger un script Scratch** proprement. Pour le Brevet, c'est décisif : un script juste n'est pas seulement un programme qui marche, c'est un enchaînement logique où chaque **bloc** a une fonction claire.

## Réviser Scratch en 3ème sans perdre de temps : plan sur 7 jours avant le Brevet

---

La meilleure façon de **réviser scratch brevet** sur **7 jours** consiste à alterner lecture de script, exercice court, correction active et reprise ciblée des erreurs. En une semaine, un élève peut retrouver les automatismes utiles au **Brevet des collèges** sans refaire tout le cours, à condition de suivre un ordre simple : *comprendre, tester, corriger, refaire sans aide*.

Jour 1, relis une **fiche de révision** très courte sur les blocs essentiels de **Scratch** : variables, conditions, boucles, coordonnées, messages. Jour 2, fais deux exercices corrigés faciles, puis note seulement les erreurs qui reviennent. Jour 3, passe à un **programme de calcul** et demande-toi vraiment *comment faire un programme de calcul avec scratch* sans recopier un modèle. Jour 4, prends une ou deux **annales scratch brevet** et entraîne-toi à lire un script avant de toucher au clavier. C'est souvent là que les points se gagnent. Un élève de **4ème** peut déjà utiliser ces exercices, mais en **3ème**, on attend plus d'autonomie, plus de justification, et une meilleure relecture des consignes.

Jour 5, refais seulement les questions ratées, sans regarder le corrigé pendant cinq minutes. Jour 6, consacre une courte séance à un mini **projet** : par exemple un quiz ou un déplacement de sprite. Si tu te demandes *comment faire un projet avec scratch*, garde-le très simple : l'objectif n'est pas de créer un jeu complet, mais de manipuler blocs, tests et variables avec sens. Jour 7, termine par un mini bilan : une lecture de script, un exercice chronométré, puis une relecture de ta feuille d'erreurs. Les meilleures **fiches de révision scratch 3ème** tiennent sur peu de pages et servent à revoir vite. Si tu veux réviser proprement à la maison, le plus efficace reste d'**imprimer** une fiche de révision et quelques exercices corrigés, puis de les refaire une dernière fois sans aide.

### Comment expliquer scratch ?

Scratch est un langage de programmation visuel. Au lieu d'écrire du code compliqué, on assemble des blocs de couleur pour donner des instructions à un personnage ou à un objet. Je le présente souvent comme un jeu de construction numérique : on crée des animations, des histoires, des quiz ou des programmes de calcul de façon simple et logique.

### C'est quoi le logiciel Scratch ?

Scratch est un logiciel et une plateforme de programmation conçus pour apprendre à coder facilement. Il permet de créer des projets interactifs avec des blocs à glisser-déposer. En 3ème, il sert souvent à travailler la logique, les variables, les boucles, les conditions et les algorithmes, notamment dans des exercices Scratch 3ème corrigés PDF.

## **Comment faire un programme de calcul avec Scratch ?**

Pour faire un programme de calcul avec Scratch, je commence par demander un nombre à l'utilisateur. Ensuite, j'utilise des variables et des blocs de calcul pour appliquer les étapes : multiplier, ajouter, soustraire ou diviser. Enfin, j'affiche le résultat. L'idée est de traduire chaque consigne mathématique en une suite d'actions simples et ordonnées.

## **Comment faire un projet avec Scratch ?**

Pour créer un projet Scratch, il faut d'abord choisir un objectif clair : animation, jeu, quiz ou programme de calcul. Ensuite, je découpe le projet en petites étapes : personnages, décors, actions, tests. Puis j'assemble les blocs progressivement. Le plus efficace est d'avancer par essais successifs, en testant souvent pour corriger les erreurs rapidement.

## **Comment expliquer simplement Scratch à un élève de 3ème ?**

À un élève de 3ème, j'explique que Scratch sert à apprendre la logique du code sans avoir à taper des lignes compliquées. On construit un programme avec des blocs qui s'emboîtent. Cela aide à comprendre les boucles, les tests, les variables et les algorithmes. C'est une excellente base avant de passer à des langages plus avancés.

## **C'est quoi le logiciel Scratch, exactement ?**

Exactement, Scratch est un environnement de programmation visuelle créé pour l'apprentissage. Il permet de commander des sprites, gérer des événements, utiliser des variables et créer des interactions. On peut s'en servir en ligne ou via une application. En collège, il est très utilisé pour relier mathématiques, algorithmique et créativité dans des activités concrètes.

## **Comment faire un projet avec Scratch sans partir dans tous les sens ?**

Pour éviter de se disperser, je conseille de définir une consigne précise, puis un résultat final très simple. Ensuite, on liste trois ou quatre fonctions maximum à programmer. On commence par le plus important, puis on teste. Cette méthode évite d'ajouter trop d'idées d'un coup et aide à finir un projet propre, compréhensible et corrigible.

## **Qu'est-ce que le mot « scratch » veut dire en informatique scolaire ?**

En informatique scolaire, Scratch désigne surtout un outil pour apprendre à programmer. Le mot renvoie au nom du logiciel, mais aussi à une manière simple d'aborder l'algorithmique. Pour un collégien, "faire du Scratch", c'est créer un programme avec des blocs visuels afin de comprendre la logique, les instructions et l'enchaînement des actions.

Un bon PDF d'exercices Scratch 3ème corrigés doit aider à s'entraîner, mais surtout à comprendre pourquoi un script fonctionne ou non. Pour progresser avant le Brevet, privilégiez des séries courtes, progressives, avec corrigés commentés, puis refaites les



exercices sans aide. La bonne méthode consiste à repérer ses erreurs sur les variables, les boucles et les conditions, puis à s'entraîner sur des sujets proches des annales pour automatiser les bons réflexes.

**[Continue sur maths-college.fr](https://maths-college.fr)**

---

Maths collège - Document pédagogique