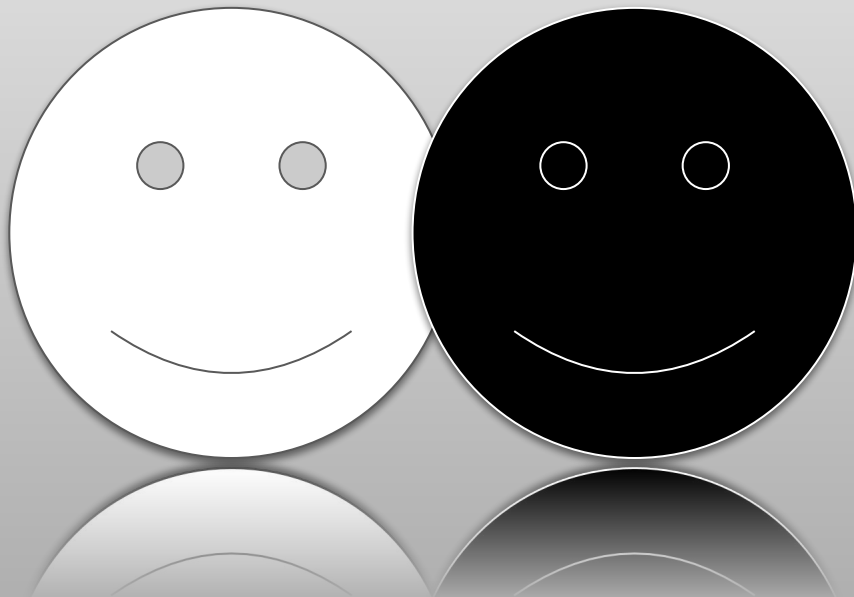




[scratch.mit.edu](https://scratch.mit.edu)



<https://www.reseau-canope.fr>

# APPARIEMENT

Retrouver les paires de cartes

Mars 2019

*Pedag O'Scratch*





## PAIRES1

### Objectifs

Pour ce programme, nous allons utiliser deux listes : *laBonnePaire* qui liste les numéros des cartes correspondantes à chaque carte. *maPaire* liste les paires de cartes effectuées par le joueur. À vous de la remplir à la main. Pour faire simple, la 1 ira avec la 2 (et donc la 2 avec la 1), la 3 avec la 4 (et donc la 4 avec la 3),...

Il s'agit donc d'abord de vider la liste *maPaire* et de la remplir de 0. Elle doit être aussi longue que *laBonnePaire*.

Ensuite Scratch explique le jeu.

Évènements

Apparence

Contrôle

Variables

supprimer tous les éléments de la liste

maPaire ▼



dire

Bonjour !

pendant

2

secondes

répéter

fois

ajouter

0

à

maPaire ▼

dire

Clique sur moi pour vérifier.

pendant

2

secondes

quand



est cliqué

longueur de

laBonnePaire ▼

dire

Mets ensemble les cartes qui vont par paires.

pendant

4

secondes

Vérifiez que vos deux listes ressemblent à cet exemple en cliquant sur le drapeau vert.

	maPaire		maBonnePaire
1	0	1	2
2	0	2	1
3	0	3	4
4	0	4	3
5	0	5	6
6	0	6	5
7	0	7	8
8	0	8	7
9	0	9	10
10	0	10	9
+ longueur 10 =		+ longueur 10 =	

Évènements

Apparence

Contrôle

Variables



```

quand [drapeau vert] est cliqué
  supprimer tous les éléments de la liste maPaire
  répéter (longueur de laBonnePaire) fois
    ajouter 0 à maPaire
  dire [Bonjour !] pendant 2 secondes
  dire [Mets ensemble les cartes qui vont par paires.] pendant 4 secondes
  dire [Clique sur moi pour vérifier.] pendant 2 secondes
  
```

## PAIRES2

### Objectifs

Pour les cartes, il s'agit d'abord de lui donner son numéro avec la variable *monNuméro*. On initialise ensuite sa taille et son costume. Le costume *Carte* correspond à une carte non appairée (on dira "solitaire" par la suite). Le bord est normal et le fond est gris. Nous verrons les autres costumes par la suite.

Une boucle va placer aléatoirement la carte sur la scène jusqu'à ce qu'elle ne touche pas le bord.

Mouvement

Apparence

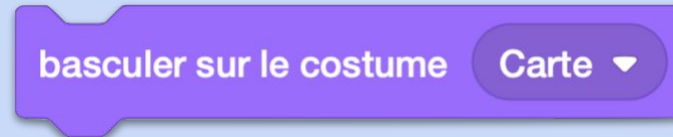
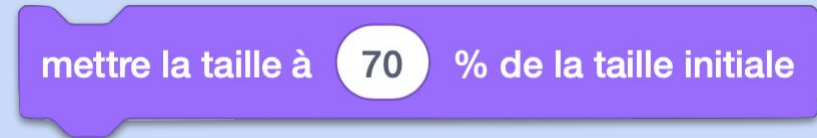
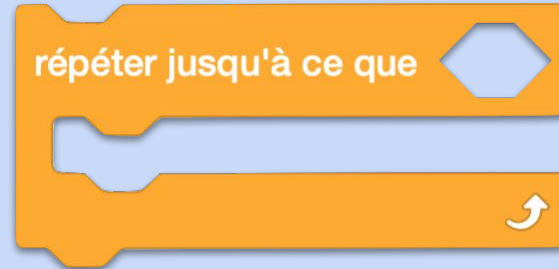
Capteurs

Événements

Contrôle

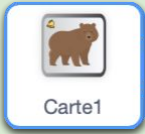
Opérateurs

Variables



Si plusieurs cartes sont les unes sur les autres, ce n'est pas grave, le joueur pourra les déplacer pour mieux les voir.

Si vous voulez, vous pouvez rajouter la condition suivante : ne pas toucher Scratch. Voire même, ne pas toucher une autre carte comme pour le jeu de Memory, mais cela semble moins utile.



```

    quand [drapeau] est cliqué
    mettre [monNuméro] à 1
    basculer sur le costume [Carte]
    mettre la taille à 70 % de la taille initiale
    aller à position aléatoire
    répéter jusqu'à ce que [non] touche le [bord] ?
    aller à position aléatoire
  
```

Mouvement	Apparence	
Évènements	Contrôle	Capteurs
Opérateurs	Variables	

## PAIRES3

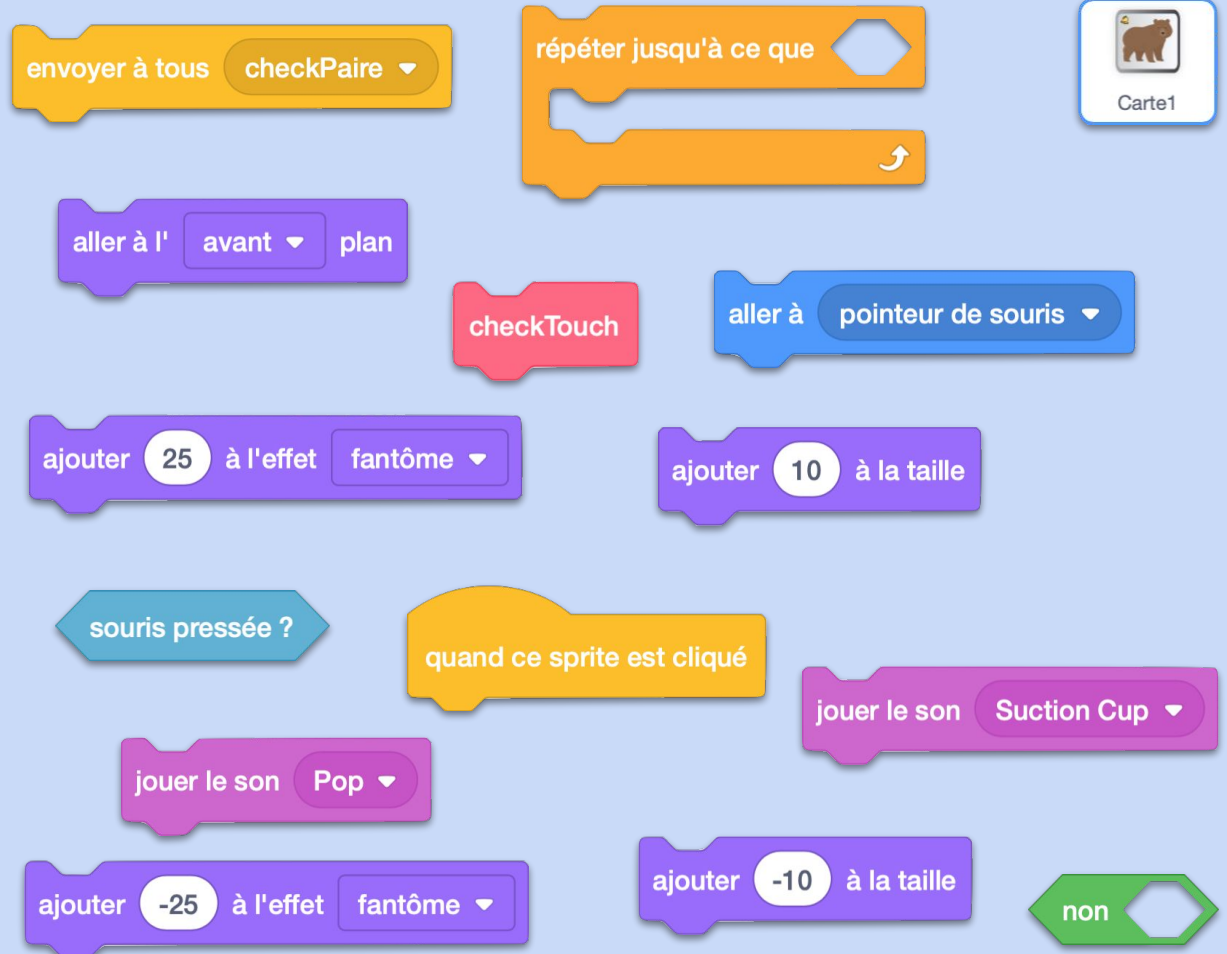
### Objectifs

Occupons-nous maintenant des clics sur les cartes. D'abord un petit effet ergonomique : son de soulèvement, la carte grossit et devient un peu transparente et finalement "passe devant" les autres cartes.

Elle va ensuite suivre le curseur de la souris jusqu'à ce que le clic soit relâché. On ajoute alors les effets ergonomiques : son de dépôt, retour à la taille et à la transparence normale.

On ajoute enfin, le bloc *checkTouch* et l'envoi du message *checkPaire*. Nous verrons plus loin de quoi il s'agit exactement.

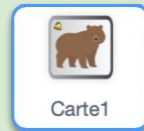
Mouvement	Apparence	Son
Événements	Contrôle	Capteurs
Opérateurs	Variables	Mes Blocs



Le bloc *checkTouch* est un bloc dont vous allez entièrement programmer les effets. Il va permettre de vérifier si la carte touche une autre carte et si c'est sa carte paire.

Le message *checkPaire* permettra de basculer sur le costume correspondant à l'état de la carte : solitaire ou appairée, aussi bien pour la carte que pour les autres cartes.

Mouvement	Apparence	Son
Événements	Contrôle	Capteurs
Opérateurs	Mes Blocs	



```

quand ce sprite est cliqué
  jouer le son Pop
  aller à l'avant plan
  ajouter 10 à la taille
  ajouter 25 à l'effet fantôme
  répéter jusqu'à ce que non souris pressée ?
    aller à pointeur de souris
  jouer le son Suction Cup
  ajouter -10 à la taille
  ajouter -25 à l'effet fantôme
  checkTouch
  envoyer à tous checkPaire
  
```



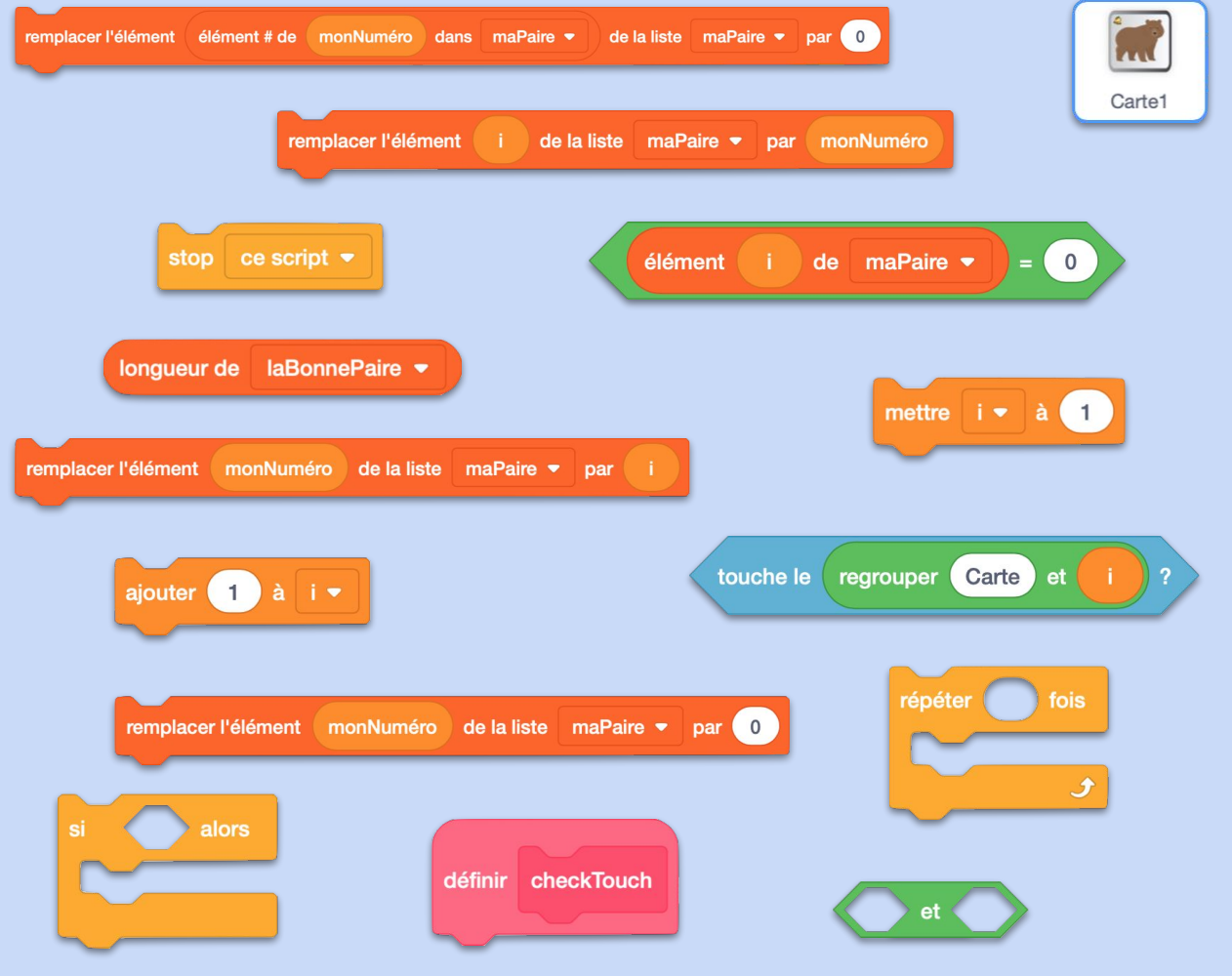
# PAIRES4

## Objectifs

Le bloc *checkTouch* va d'abord enlever la carte touchée de la liste *maPaire* si elle s'y trouve. Puis enlever la carte éventuellement appairée avec elle de cette même liste.

Ensuite on lance une boucle dans laquelle la variable locale *i* permettra de vérifier si la carte en touche une autre. *i* prendra pour valeur les nombres de 1 jusqu'au nombre total de cartes.

Si c'est le cas et que celle-ci n'est pas déjà appairée avec une autre, alors on fait les changements dans la liste *maPaire* pour indiquer que ces deux cartes sont maintenant appairées. On s'arrête dès qu'on en trouve une, même si elle touche en fait plusieurs cartes.



The image shows a Scratch script for a game called "PAIRES4". The script is as follows:

- remplacer l'élément élément # de monNuméro dans maPaire de la liste maPaire par 0
- remplacer l'élément i de la liste maPaire par monNuméro
- stop ce script
- longueur de laBonnePaire
- mettre i à 1
- remplacer l'élément monNuméro de la liste maPaire par i
- ajouter 1 à i
- remplacer l'élément monNuméro de la liste maPaire par 0
- si alors
- defini checkTouch
- et
- regrouper Carte et i ?
- répéter fois

The script is displayed on a light blue background. A small icon of a bear is in the top right corner, labeled "Carte1". The script is composed of several blocks: orange blocks for list manipulation, a yellow stop block, a green loop block, a blue loop block, a blue conditional block, a yellow loop block, and a pink block for defining a function.

Contrôle

Capteurs

Opérateurs

Variables

Mes Blocs

On peut noter, qu'une carte ne se touche jamais elle-même.

Si plusieurs cartes se touchent, c'est celle avec le plus petit nombre et qui n'est pas déjà appairée qui sera considérée.

Attention au nommage des cartes, parce que l'on reconstruit par une instruction "regroupe X et Y" son nom.

C'est d'ailleurs une des petites astuces : on peut mettre une variable dans l'instruction "touche le Lutin" alors qu'il y a un menu déroulant par défaut. C'est à la fois pratique et source potentiel de bugs difficiles à détecter.



Carte1

```

définir checkTouch
  remplacer l'élément monNuméro de la liste maPaire par 0
  remplacer l'élément élément # de monNuméro dans maPaire de la liste maPaire par 0
  mettre i à 1
  répéter longueur de laBonnePaire fois
    si touche le regrouper Carte et i ? et élément i de maPaire = 0 alors
      remplacer l'élément monNuméro de la liste maPaire par i
      remplacer l'élément i de la liste maPaire par monNuméro
    stop ce script
  ajouter 1 à i
  
```

## PAIRES5

### Objectifs

Lorsque l'on clique sur Scratch, il va d'abord vérifier que toutes les cartes sont appairées. Si ce n'est pas le cas, on affiche un message d'erreur. Sinon, on met la nouvelle variable *gagné* à 1. Le message *afficheOK* (voir fiche suivante) va donner la valeur 1 à la variable *gagné* si tous les cartes sont appairées correctement sinon 0. Si c'est 1, on félicite le joueur et tous les scripts s'arrêtent. Sinon, on envoie le message *checkPaire* qui va remettre les cartes aux bonnes couleurs pour continuer le jeu.

Apparence

Son

Évènements

Contrôle

Opérateurs

Variables

The image shows a Scratch script for a game called 'PAIRES5'. The script is composed of several blocks:

- When clicked:** A yellow block labeled 'quand ce sprite est cliqué'.
- Control:** A yellow block 'stop tout'.
- Logic:** A yellow 'si...alors...sinon' block. The 'alors' part contains a red block 'maPaire contient 0 ?'. The 'sinon' part contains a purple 'dire' block 'Au moins deux cartes ne sont pas appairées.' pendant 2 secondes.
- Sound:** A purple block 'jouer le son Cheer'.
- Control:** A yellow block 'envoyer à tous checkPaire'.
- Control:** A purple 'dire' block 'Bravo c'est réussi !' pendant 5 secondes.
- Control:** An orange block 'mettre gagné à 1'.
- Control:** A green block 'gagné = 1'.
- Control:** A yellow block 'envoyer à tous afficheOK et attendre'.
- Control:** A purple 'dire' block 'Au moins 2 paires ne sont pas correctes.' pendant 2 secondes.

The Scratch logo and 'Scratch le ...' are visible in the top right corner.

Nous aurions pu utiliser l'opérateur "=" pour comparer deux listes, mais ce n'est pas du tout fiable pour l'instant. Pour cela, Scratch convertit une liste en une chaîne de caractères. Mais parfois, suivant qu'il considère que ces données sont des nombres ou du textes, il rajoute des espaces entre les termes. Du coup, une liste (1,2,3,4) est parfois interprétée "1234" et parfois "1 2 3 4" et donc ne sont pas considérées égales, sans que l'on comprenne bien comment il choisit l'un ou l'autre.

À retenir : sous votre code, il y a un autre code auquel vous n'avez pas toujours accès et qui peut aussi être bogué.

Apparence

Son

Événements

Contrôle

Opérateurs

Variables



Scratch le ...

```

quand ce sprite est cliqué
si maPaire contient 0 ? alors
  dire Au moins deux cartes ne sont pas appairées. pendant 2 secondes
sinon
  mettre gagné à 1
  envoyer à tous afficheOK et attendre
  si gagné = 1 alors
    jouer le son Cheer
    dire Bravo c'est réussi ! pendant 5 secondes
    stop tout
  sinon
    dire Au moins 2 paires ne sont pas correctes. pendant 2 secondes
    envoyer à tous checkPaire
  
```

## PAIRES6

### Objectifs

- Pour le message *checkPaire*, on affiche le costume *Carte* (fond gris) si la carte n'est pas appairée, et le costume *Paire* (fond noir) si elle l'est.
- Pour le message *afficheOK*, il s'agit de vérifier si l'élément *monNuméro* de la liste *maPaire* est le même que l'élément *monNuméro* de la liste *laBonnePaire*. Si c'est le cas, on affiche le costume *Carte vrai* (bords verts), sinon on affiche le costume *Carte faux* (bords rouges) et la variable *gagné* prend la valeur 0.

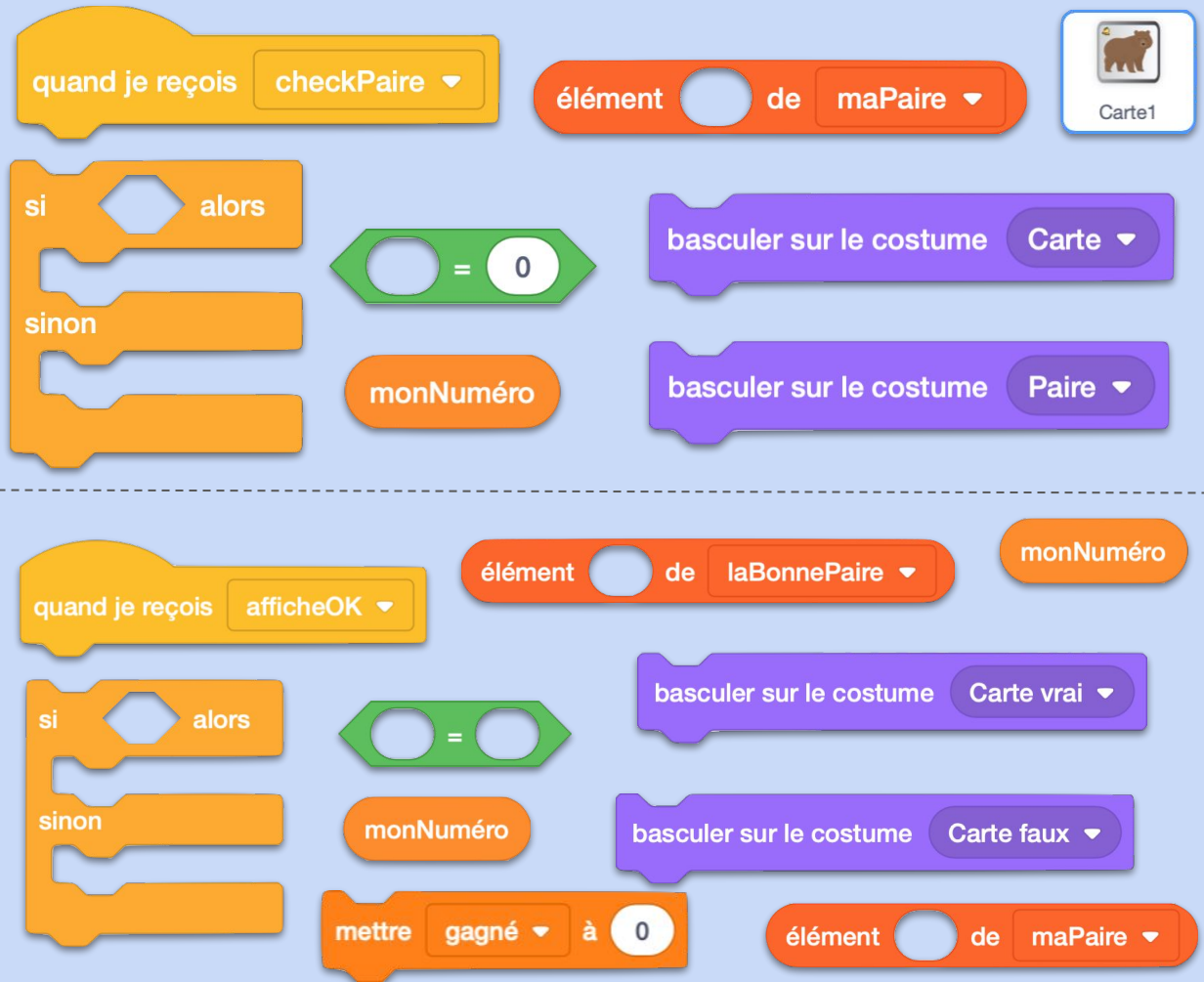
Apparence

Évènements

Contrôle

Opérateurs

Variables



## PAIRES6 ok

### Commentaires

Le message *afficheOK* est lancé avec l'option "attendre" (voir fiche 5), c'est pour être sûr que tous ces scripts dans toutes les cartes seront terminés avant que le script de Scratch qui les a lancés se poursuive. Il en suffit d'un qui mette *gagné* à 0 pour que Scratch sache que la partie n'est pas terminée.

Il ne vous reste plus qu'à dupliquer cette Carte1.

Pensez à changer l'affectation de *monNuméro* sur la fiche d'activité 2.

Répétez pour autant de paires de cartes que souhaité.

Modifiez manuellement si besoin la liste *laBonnePaire* pour indiquer la solution.

La double information (X est lié à Y et Y est lié à X) est nécessaire pour notre algorithme.

Apparence

Événements

Contrôle

Opérateurs

Variables

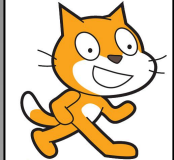
```
quand je reçois checkPaire
si élément monNuméro de maPaire = 0 alors
  basculer sur le costume Carte
sinon
  basculer sur le costume Paire
```



Carte1

```
quand je reçois afficheOK
si élément monNuméro de maPaire = élément monNuméro de laBonnePaire alors
  basculer sur le costume Carte vrai
sinon
  basculer sur le costume Carte faux
  mettre gagné à 0
```

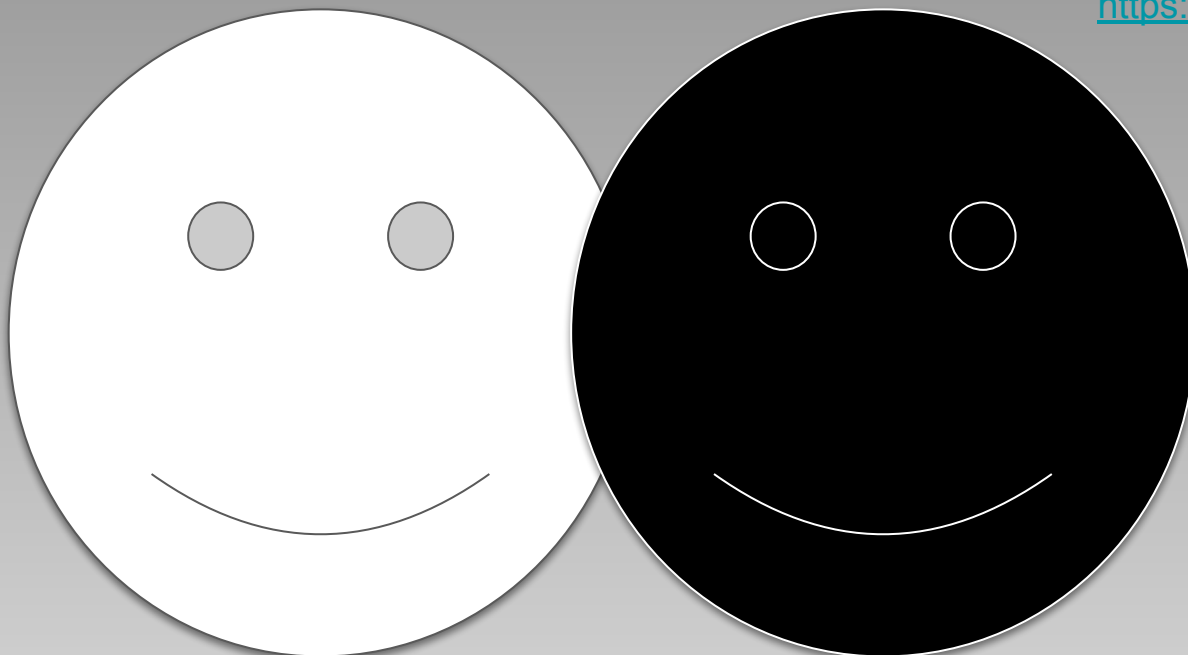




[scratch.mit.edu](https://scratch.mit.edu)



<https://www.reseau-canope.fr>



*Pedag O' Scratch*

