

Projet API

Jeu Dobble

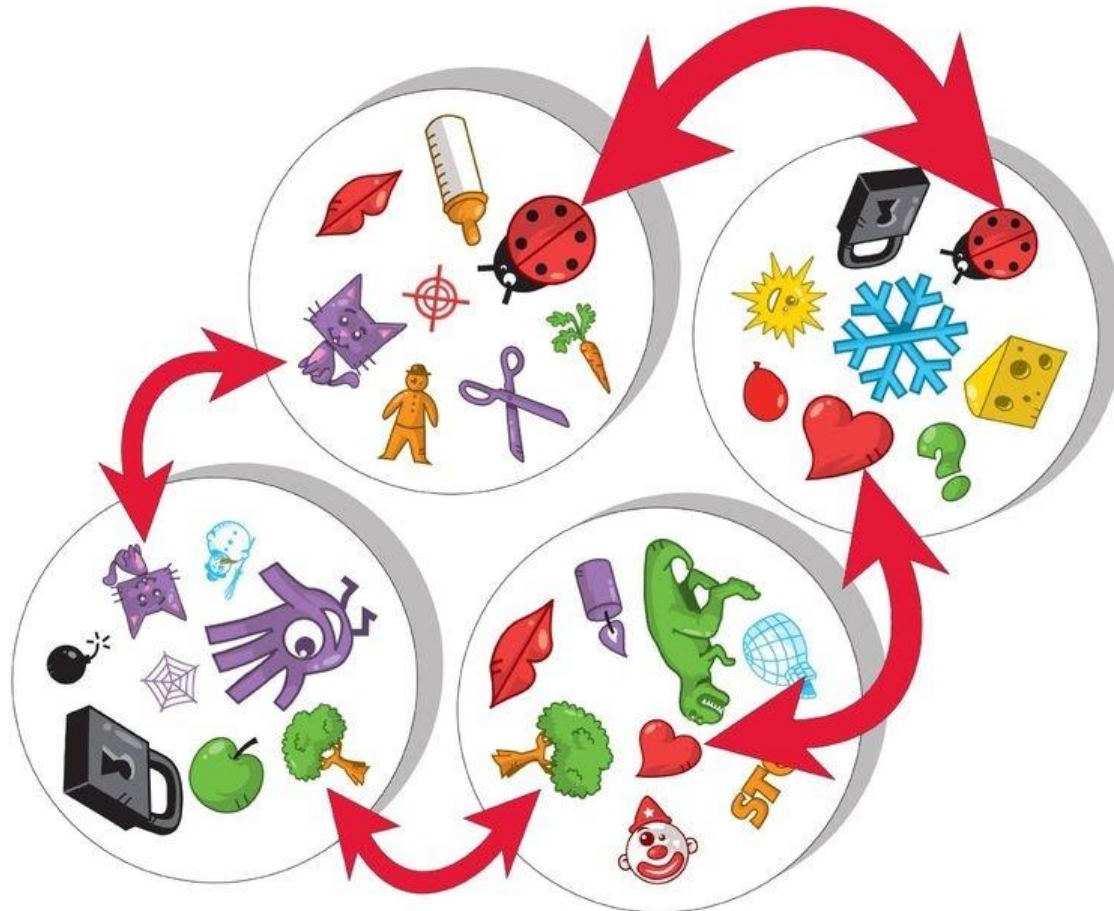


Le principe

57 cartes, 8 icônes par cartes

Entre chaque couple de carte, UNE SEULE ICÔNE COMMUNE

Trouver l'icône commune le plus vite possible



Version interactive

Cliquer dans la carte du haut sur l'icône commune avec la carte du bas.

Bonne réponse => Score +1

Temps +3 s

Mauvaise réponse => Score inchangé

Temps -3s

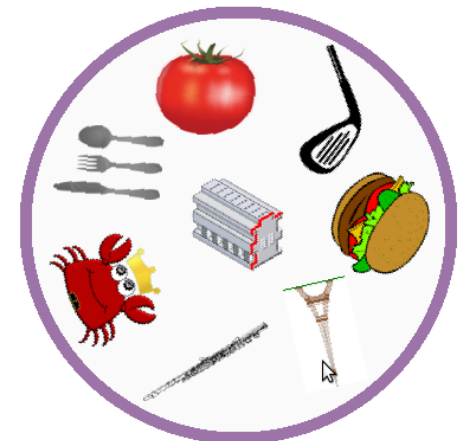
Temps de départ: 30s

RICM3-Dobble

Score 7

Temps restant

28s



Abstraction et généralisation

Fichier pg27.txt

```
57 8
0 1 2 3 4 5 6 7
0 8 9 10 11 12 13 14
1 8 15 16 17 18 19 20
.
.
.
3 12 19 22 30 33 39 42
3 13 18 21 28 35 40 45
7 11 16 22 28 34 37 48
6 10 17 21 29 33 38 48
```

La première ligne contient le nombre de cartes
et le nombre d'icône

Chaque ligne suivante représente une carte

Les entiers représentent des icônes

Deux lignes ont un unique entier en commun

Autant d'icônes que de cartes

(les Maths derrière: plan projectif fini)

p22.txt: 7 cartes, 3 icônes par carte

p23.txt: 13 cartes, 4 icônes par carte

p24.txt: 21 cartes, 5 icônes par carte

p25.txt: 31 cartes, 7 icônes par carte

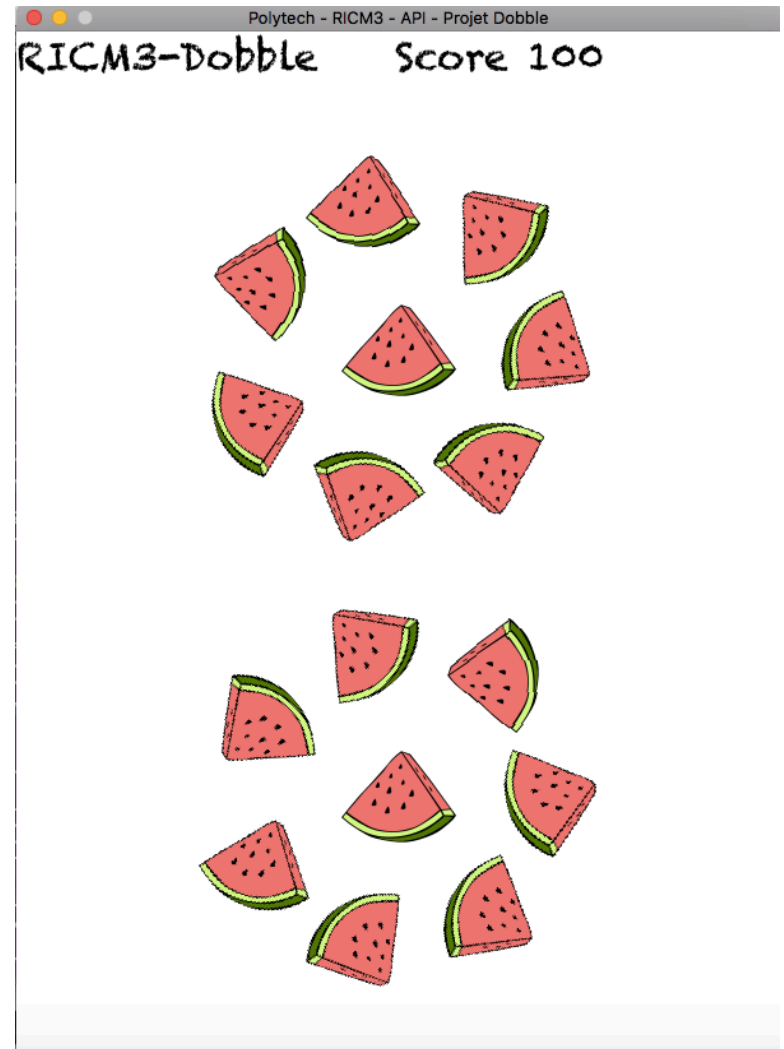
p27.txt: 57 cartes, 8 icônes par carte

p28.txt: 73 cartes, 9 icônes par carte

p29.txt: 91 cartes, 10 icônes par carte

Votre point de départ

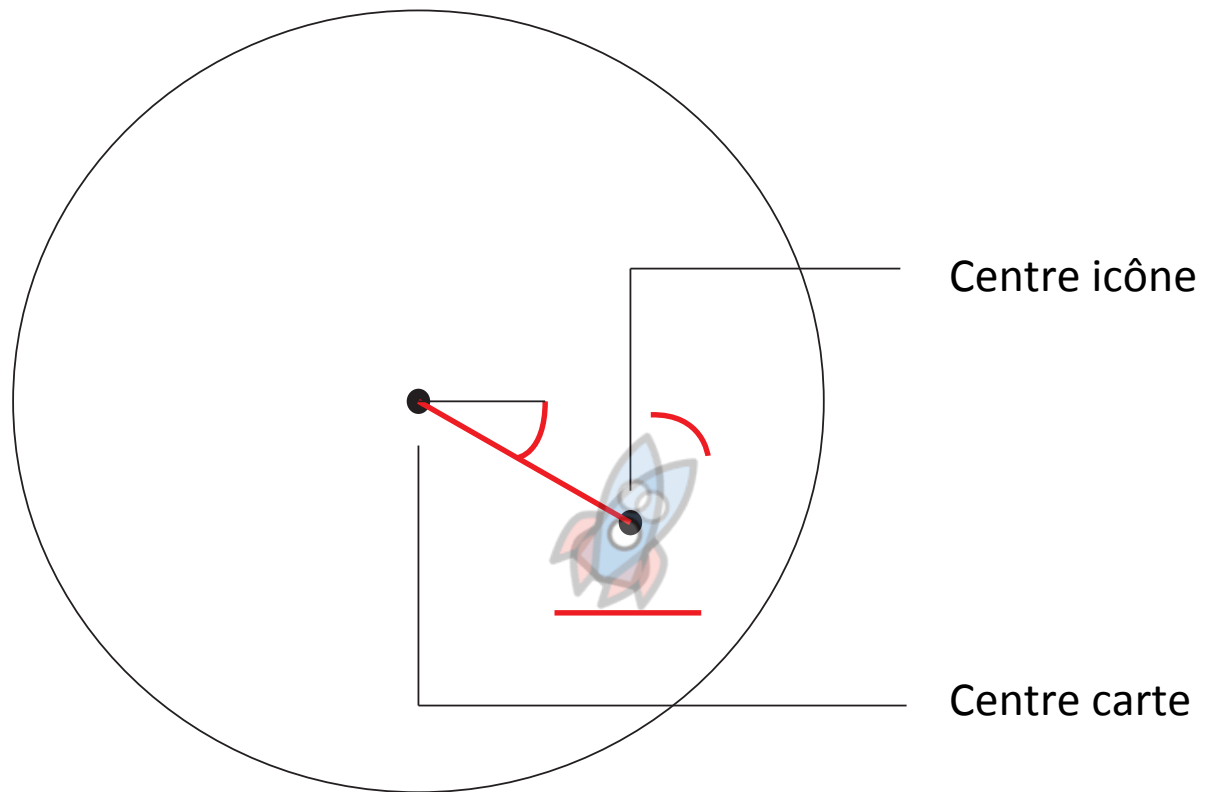
- Archive Dobble.zip
 - Projet C
 - Librairies SDL2, SDL2_Image, SDL2_TTF
- Lecture icônes
- Affichage texte et icônes
- Détection position souris
- Détection click souris
- Compte-à-rebours



Votre objectif final

- Implémenter plusieurs versions du jeu
 - Différents nombres de cartes et d'icônes
 - Positionnement des icônes simple ou complexe

La position des icônes



La position des icônes

- 4 paramètres (flottants) par icônes:
 - Distance centre carte -> centre icône
 - Angle centre carte -> centre icône (sens horaire)
 - Taille icône (relative)
 - Angle rotation icône autour de son centre

Où trouver des icônes?

- openclipart.org



Stockage des icônes dans une image

- Utilitaire montage de ImageMagick

```
montage -background none OPEN_CLIP_ART/* -geometry 90x90 AssemblageIcones.png
```

- Agence des images sous forme matricielle après les avoir réduites à la taille 90x90

Stockage des icônes dans une image



Matrice 3x3



Matrice 10x18

Les étapes obligatoires

- Compiler le programme de départ
 - Installation de librairies
 - Problèmes locaux à vos machines
 - ...
- Comprendre ce programme
 - faire un diagramme des procédures et fonctions faisant les appels
 - faire des modifications ponctuelles pour assurer la bonne compréhension de tous les éléments

Le programme de départ

- Gestion graphique bas-niveau:
 - `graphics.c, h`
 - Très probablement rien à modifier à ces fichiers
 - Contient tous les appels à la librairie SDL
- Programme principal
 - `dobble.c, h`
 - C'est sur ces fichiers que vous travaillerez
 - "Etanches" à la librairie SDL

Librairie graphique SDL

- Le principe:
 - Initialisation
 - Boucle infinie
 - Attente événement utilisateur
 - Réponse événement
 - Affichage:
 - contient du texte et des icônes
 - Tout changement implique un redessin complet
 - Le nouvel affichage est calculé en arrière-plan avant d'être présenté instantanément en premier plan

Les étapes obligatoires (suite)

- la sélection d'icône (picking):
 - À partir des coordonnées x,y de la souris, trouver le numéro de l'icône cliquée par l'utilisateur
 - Pour des positions quelconques des icônes
- Le choix des positions des icônes
- La lecture des fichiers de cartes
- La lecture des images d'icône
- Gérer le compte-à-rebours et le score
- Gérer les étapes du jeu
 - Accueil
 - Déroulement
 - Fin

Des étapes optionnelles

- Chercher des positionnements complexes des icônes sur la carte
 - Problème potentiel de recouvrement des icônes
- Personnaliser les icônes
- Mesurer des statistiques de meilleur score en fonction du positionnement choisi

Forme du rendu

- Une archive contenant votre programme
- Un rapport (max. 3 pages) décrivant
 - Les solutions choisies
 - Les difficultés rencontrées
 - Le temps dévolu au projet et sa répartition
 - ...