1 Instructions générales

Créez un nouveau projet en langage Java, nommé selon le schéma "TP5b_numéro", où "numéro" est votre numéro d'étudiant.

Assurez-vous que la "minSdk" soit bien 19 ! On peut soit choisir ce paramètre à la création du projet, soit le changer dans le `build.gradle` a posteriori. Un projet qui ne compile pas, ou qui ne peut pas être exécuté sur les appareils prêtés (Sdk = 19) recevera 0.

Une fois le projet terminé, exportez votre projet (`File` → `Export` → `Export to Zip File`), et rendez ce fichier sur Eprel.

La Section 2 constitue le sujet du TP. La Section 3 détaille la procédure de notation. On pourra s'y référer pendant la séance pour prendre connaissance des modalités d'évaluation.

2 Midterms racer

L'objectif de ce TP est de réaliser une implémentation du jeu **Midterms racer**, qui fait carton plein aux États-Unis à l'approches des élections de mi-mandat. L'état du jeu, au départ, est représenté en Figure 1.

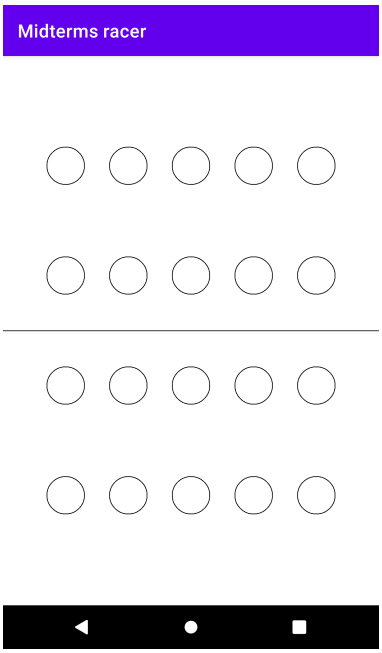


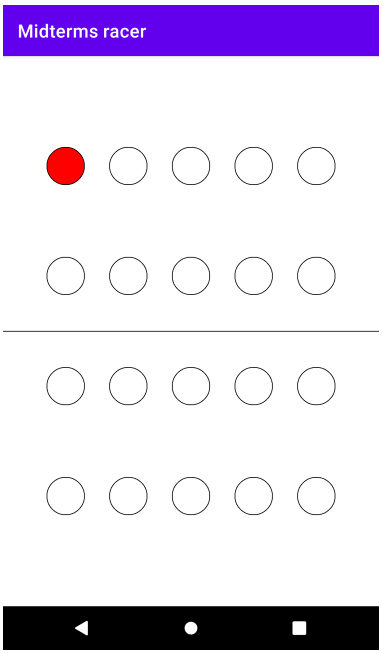
FIGURE 1 – Aperçu de l'écran de départ du jeu.

Dans **Midterms racer**, deux camps s'affrontent : les Républicains (en rouge) et les Démocrates (en bleu).

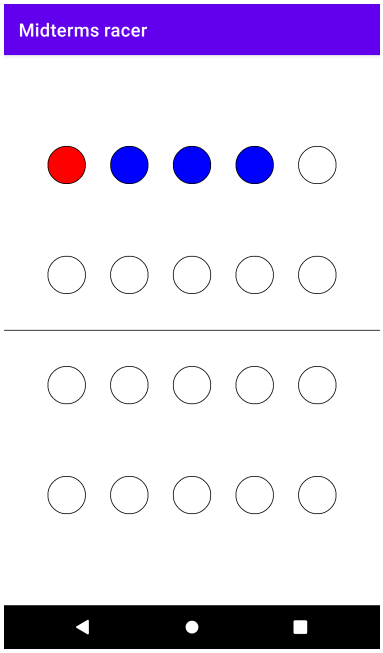
L'idée du jeu est la suivante : l'interface contient vingt sièges (représentés par des cercles), qui constituent le Sénat. Tout l'enjeu des élections de mi-mandat est de répartir ces sièges entre les deux camps.

Par ailleurs, l'écran est divisé en deux parties : celle du haut appartient au camp républicain, et

celle du bas au camp démocrate. Notez bien que les sièges sont communs, peu importe dans quelle partie de l'écran ils se trouvent.
Quand un joueur ou une joueuse clique sur sa partie d'écran, il ou elle récupère un siège. La Figure 2 représente l'état du jeu, quand, depuis la position initiale (Figure 1), les Républicains cliquent une fois dans leur partie d'écran (Figure 2a) puis les Démocrates cliquent trois fois dans la leur (Figure 2b). Les sièges sont donc attribués ligne par ligne, en partant du coin en haut à gauche.



(a) ...les Républicains cliquent une fois dans leur camp...



(b) ...puis les Démocrates cliquent trois fois dans leur camp.

FIGURE 2 – Depuis la position initiale...

L'objectif du jeu est de récupérer plus de sièges que son adversaire. Par exemple, dans la situation décrite en Figure 3, les Démocrates possèdent 10 sièges, et les Républicains seulement 9. Les Démocrates gagnent donc s'ils arrivent à récupérer le dernier siège - autrement, ce sera une partie nulle.

2.1 Création du `Plateau`

Question 1 Durant la partie, on cherche à éviter les rotations d'écran.
On ajoutera donc la ligne suivante dans le Manifest, comme attribut de `MainActivity` :

```
<android:screenOrientation=''portrait''>
```

Question 2 Créez une class `Plateau` (vide pour l'instant) qui étend `View`.
On créera un nouveau `Plateau` dans `MainActivity::onCreate()`, qu'on déclarera comme unique élément de notre layout via `setContentView()`.

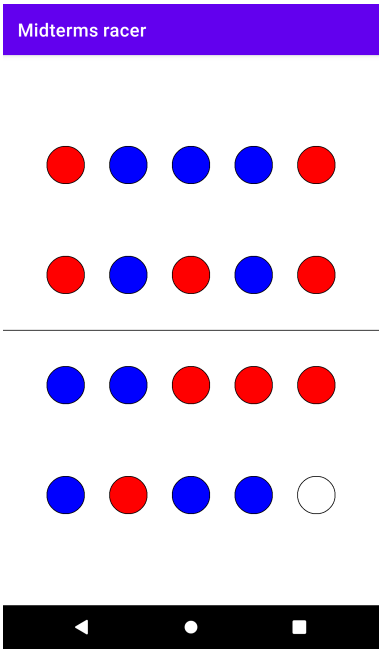


FIGURE 3 – Les Démocrates seront victorieux s'ils récupèrent le dernier siège. Dans le cas contraire, chaque camp aura 10 sièges, et la partie se terminera par un ex aequo.

Question 3 Faites en sorte qu'un `Plateau`, en se dessinant, trace une ligne noire horizontale coupant l'écran en deux : ce sera la ligne de séparation entre les deux camps.

Question 4 Recopiez les méthodes `Plateau::computeCenters()` et `Plateau::computeRadius()` fournies ci-dessous. La première retourne un `float[20][2]`, qui contient la liste des positions (sous forme de tableau de taille 2) des centres des 20 sièges, dans l'ordre de remplissage attendu (c'est-à-dire que `center[0]` correspond aux coordonnées du sièges en haut à gauche, etc. ; cet ordre est représenté en Figure 4). La deuxième méthode retourne le rayon des sièges.

À l'aide de ces méthodes, modifiez votre code pour qu'en se dessinant, un `Plateau` fasse apparaître les 20 sièges, avec les bons centres et les bons rayons. On utilisera le même `Paint` que pour la Question 3. Pour rappel, les dimensions d'une `View` doivent être calculées dans `Plateau::onDraw()`, et pas dans le constructeur (puisque les dimensions ne sont pas encore fixées à ce moment-là).

```
private float[][] computeCenters(){
        float[][] centers = new float[20][2];
        float w = getWidth(), h = getHeight();

        for(int i = 0; i<20; i++){
            centers[i][0] = ((i%5) + 1)*w/6;
            centers[i][1] = (i/5 + 1)*h/5;
        }
        return centers;
    }
```

```
    private float computeRadius(){
        return Math.min(getWidth(),getHeight())/20;
    }
```

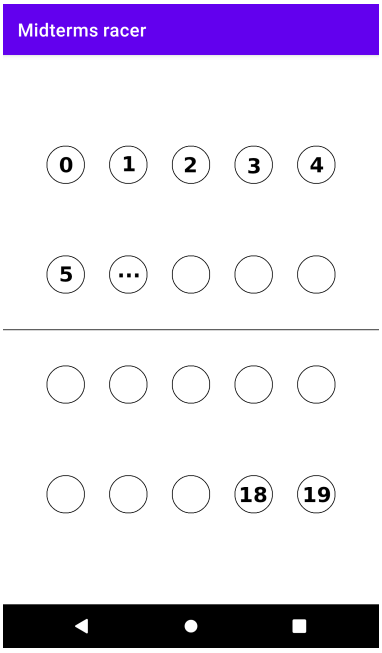


FIGURE 4 – Les indices des différents sièges dans le tableau `centers`.

2.2 Moteur de jeu

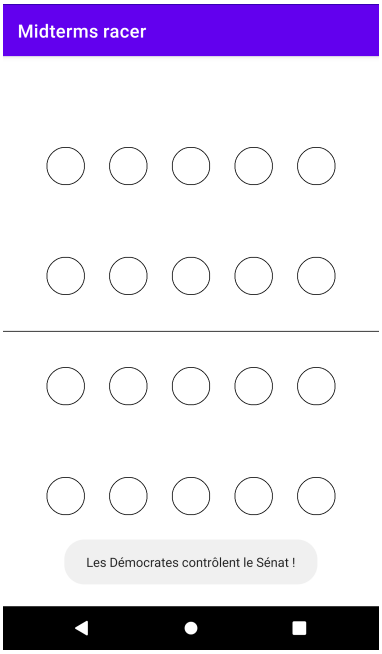
Il nous reste maintenant à faire en sorte qu'un clic sur l'écran permette au camp correspondant de gagner le prochain siège non-attribué.

Question 5 Déclarez un tableau d'entiers de taille `20` qui servira à stocker l'attribution des sièges. On pourra considérer qu'un `0` dans une case signifie que le siège n'est pas encore attribué, tandis qu'un `1` signifie que ce siège appartient aux Républicains, alors qu'un `2` représente le fait que ce siège est acquis aux Démocrates. Au début, tous les sièges sont neutres.
Modifiez une dernière fois `Plateau::onDraw` pour que les sièges des Républicains soient coloriés en `Color.RED`, et ceux des Démocrates en `Color.BLUE`. On créera pour cela deux nouveaux `Paint`.

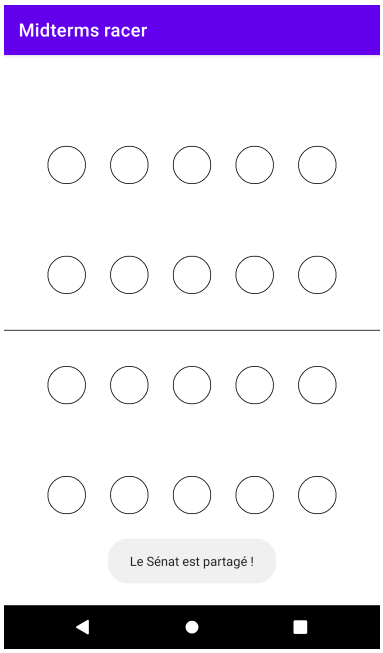
Question 6 Implémentez `Plateau::onTouchEvent()` pour qu'un `MotionEvent.ACTION_DOWN` sur la moitié supérieure de l'écran (respectivement, sur la moitié inférieure de l'écran) fasse passer chez les Républicains (respectivement, les Démocrates) le premier siège qui n'est pas encore attribué.

Pour rappel, un exemple du comportement attendu est précisé en Figure 2.

Question 7 Pour finir, implémentez la méthode `Plateau::checkWinner()` qui vérifie la condition de victoire, et qui sera appelée quand le dernier siège sera attribué. Si un camp a strictement plus



(a) ...les Démocrates gagnent s'ils sont les premiers à cliquer dans leur camp...



(b) ...autrement, c'est match nul.

FIGURE 5 – Depuis la position décrite en Figure 3...

de sièges que l'autre, on veut afficher un `Toast` "Les Républicains contrôlent le Sénat !" ou "Les Démocrates contrôlent le Sénat !" (en fonction du camp victorieux). Si le match est nul, on veut afficher un `Toast` "Le Sénat est partagé". Dans tous les cas, on veut que le jeu soit réinitialisé.

Par exemple, la situation qui suit une victoire des Démocrates est représentée en Figure 5a, et celle qui suit un match nul est décrite en Figure 5b.

3 Notation

Au moment de corriger le TP de votre camarade, téléchargez le fichier zip qui vous a été attribué sur Eprel, et importez le projet dans Android Studio (`File` → `New` → `Import Project`). Merci de ne pas chercher à savoir à laquelle ou auquel de vos camarades appartient le numéro d'étudiant présent dans le nom du projet.

Pour la notation, on se contentera de tester l'application sur un appareil, sans en regarder le code. Un code qui ne compile recevra une note de 0. La liste des éléments à vérifier sur l'application que vous avez à noter est donnée en Figure 6. Pour chaque critère, on attribuera :

— 0, si le critère n'est pas du tout respecté.
— La moitié des points si le critère est partiellement respecté.
— L'intégralité des points si le critère est parfaitement respecté.

Critère	**Note**
Au lancement...	
...l'écran est séparé en deux parts égales par une ligne noire horizontale	2
...les vingt sièges sont bien tracés en noir à l'écran	5
Après deux ou trois clics sur la moitié supérieure de l'écran...	
...chaque clic remplit un siège en rouge, en partant du haut à gauche	2
En cliquant ensuite deux ou trois fois sur la moitié inférieure de l'écran...	
...chaque clic remplit le siège suivant en bleu	2
En continuant la partie, en ayant cliqué au total 12 fois en haut et 8 fois en bas...	
...le message "Les Républicains ont gagné le Sénat !" ou le message "Les Républicains contrôlent le Sénat !" apparaît quelques secondes à l'écran	2
...les sièges se vident	2
Après une nouvelle partie, en cliquant au total 9 fois en haut et 11 fois en bas...	
...le message "Les Démocrates ont gagné le Sénat !" ou le message "Les Démocrates contrôlent le Sénat !" apparaît quelques secondes à l'écran	2
Faites une nouvelle partie en cliquant, en alternant, 10 fois en haut et 10 fois en bas...	
...le message "Le Sénat est partagé !" apparaît quelques secondes à l'écran	3

FIGURE 6 – Liste des critères à vérifier, et nombre de points associés.