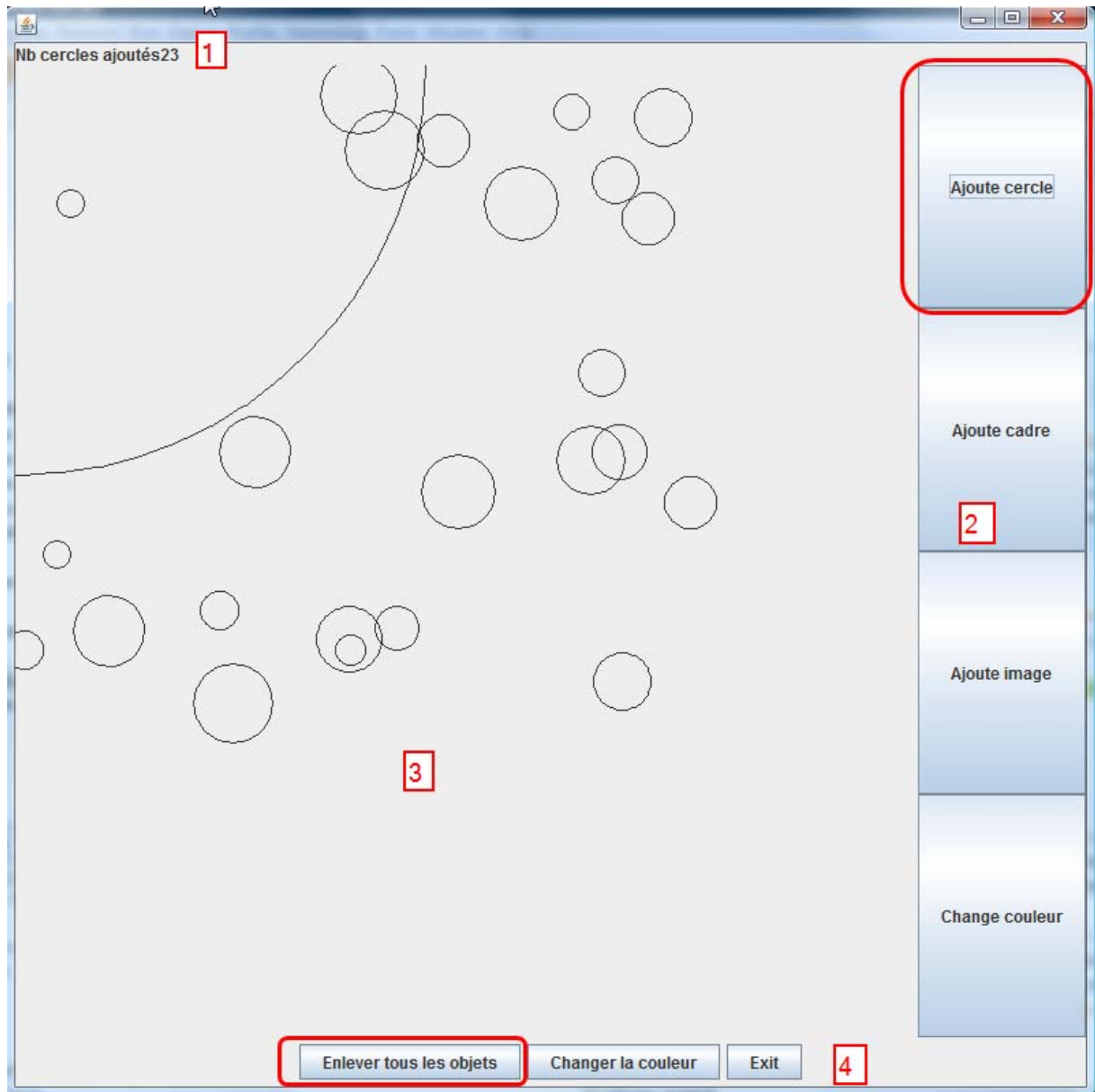


## Examen P00 – Partie 2

On veut reconstruire l'interface graphique suivante :



La partie centrale de cette fenêtre (partie 3) est une instance de la classe `PanelDessin` (code fourni page suivante). On vous demande de créer la classe `MaFenetre` qui construit cette interface graphique. Vous vous inspirerez des exemples des supports de cours.

**IMPORTANT :** on n'attend pas un code super détaillé, on supposera que les imports sont ok, pas besoin de mettre des accesseurs ou des modifieurs, etc. L'important est que le correcteur comprenne que vous avez compris !

1. On supposera que la classe `MaFenetre` dérive de la classe `JFrame`, vous construirez l'ensemble de l'interface graphique dans le constructeur sans paramètre de la classe `MaFenetre`. La méthode `main` est la suivante :

```

public static void main(String[] args) {
    MaFenetre f = new MaFenetre();
}

```

2. L'interface graphique contient 4 JPanels, à vous de construire les panels 1, 2 et 4. Pensez à bien spécifier quel LayoutManager vous utilisez et comment vous l'initialisez. Pensez à ajouter de la bonne manière les panels à l'instance de la classe MaFenetre.
3. Vous ajouterez un écouteur au bouton « Ajoute Cercle », qui appelle la méthode addObjetDessinable de la classe PanelDessin, écrivez complètement le code de cet écouteur.
4. Même question pour le bouton « Enlever tous les objets », d'après vous quelle méthode de la classe PanelDessin appelle-t-il ?

```

public class PanelDessin extends JPanel {
    private ArrayList<Dessinable> listeObjetsDessinables = new ArrayList<Dessinable>();

    public PanelDessin(int largeur, int hauteur) {
        setPreferredSize(new Dimension(largeur, hauteur));
    }

    /** Ajoute un objet Dessinable à la liste des objets, puis l'appel à
     * repaint provoque met à jour le contenu du JPanel
     * @param o : l'objet Dessinable que l'on rajoute.
     */
    public void addObjetDessinable(Dessinable o) {
        listeObjetsDessinables.add(o);
        repaint();
    }

    @Override
    public void paintComponent(Graphics g) {
        // On efface le contenu de la fenêtre
        super.paintComponent(g);

        // Puis on dessine l'ensemble des objets
        for (Dessinable o : listeObjetsDessinables) {
            o.dessiner(g);
        }
    }

    /** Remet à zéro la liste des objets à dessiner */
    void effaceTousLesObjets() {
        listeObjetsDessinables = new ArrayList<Dessinable>();
        repaint();
    }
}

```