

## PoP Série 4

**H1 : Usage de **GTKmm 4** pour l'interface graphique utilisateur /  
Création d'une fenêtre et dessin avec Cairo**

**H2 : MOOC [MOOC Introduction à la programmation orientée objet \(en C++\)](#)**

**Série semaine 4: héritage** (exercice 15 hors programme du cours)

---

### Usage de **GTKmm 4** pour l'interface graphique utilisateur : Création d'une fenêtre et dessin avec Cairo

**Exercice 1.** (niveau 0) : **Création de votre première fenêtre avec GTKmm 4**

Le document séparé introduit les noms de fichier en-tête et la syntaxe de la commande de compilation dans geany ainsi qu'un fichier Makefile qu'on pourra adapter par la suite.

**Exercice 2.** (niveau 0) : **Création de votre première application interactive avec GTKmm 4**

Cet exemple comporte deux modules et illustre la notion d'événement, de *signal* et de *signal handler* produisant une réaction souhaitée quand on clique sur un bouton.

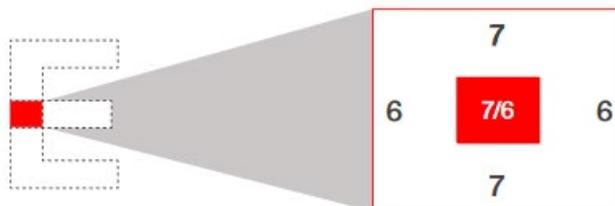
**Exercice 3** (niveau 0) : **Création de votre premier dessin avec GTKmm 4**

Cet exemple comporte deux modules ; il introduit l'objet DrawingArea pour lequel on redéfinit le signal handler `on_draw()` qui redessine la fenêtre.

**Exercice 4** (niveau 1) : **dessiner le Logo de l'EPFL =>**



Reprendre et compléter le code de l'exercice 3 pour dessiner ce logo. Il suffit de jouer sur l'épaisseur du trait pour avoir la bonne proportion. Attention la barre verticale du E et du F est plus large que les autres épaisseurs de trait (dans un rapport 7/6).



Pour l'arc du P on utilisera la méthode `arc` qui permet aussi de dessiner des cercles en jouant avec les paramètres : `arc(x, y, radius, start_angle in rd, end_angle in rd)`

(x,y) est le centre de l'arc ; les angles sont exprimés en radians. De plus, du fait de la convention d'axes, les angles croissent dans le sens des aiguilles d'une montre.