

Conception Objets avancée

Giuseppe Lipari

February 1, 2019

1 Instructions

Vous devez rendre sur gitlab le code demandé avec un fichier README qui contient :

- Vos noms ;
- Pour chaque question :
 - Si vous avez réussi à coder les fonctionnalités demandées
 - La liste de tests de régression correspondants à la question

Vous avez **deux** séances à disposition pour rendre ce TP.

2 TP 3: Inheritance sur les formes

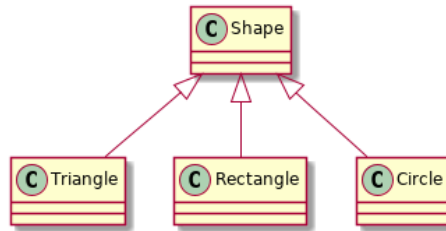
Le but de ce TP est de 1) s'entraîner avec le concept d'héritage public, et 2) utiliser une bibliothèque C++ existante. Il faut écrire un programme pour créer, dessiner et manipuler des formes géométriques sur l'écran.

2.1 La bibliothèque CImg

Pour visualiser les formes, on utilise la librairie *CImg* <http://cimg.eu/>. La librairie est composée d'un seul fichier `CImg.h` qu'il faut inclure dans notre programme. Un exemple de programme est fourni dans le fichier `example_cimg.cpp`.

La librairie permet ainsi de recevoir des commandes par la souris et par le clavier : la méthode `main_disp.wait()` ; attend un "évènement", c'est à dire un mouvement de la souris, un clic du bouton de la souris, ou la pression d'un touche. On peut ensuite examiner de quel "évènement" il s'agit avec des méthodes comme `main_disp.is_key(cimg::keyA)` ;, ou `main_disp.button()`, ou encore `main_disp.mouse_x()` et `main_disp.mouse_y()`.

La documentation de la librairie est fournie dans `docs/CImg_reference.pdf`.



2.2 Q1 : Les formes

Vous devez concevoir une hiérarchie de classes **Shape** (forme). Voici le diagramme :

Les formes ont une position (x, y) sur le plan 2D. Elles peuvent être déplacées (changer la position) avec la fonction `move()` et ils peuvent être dessinées avec la fonction `draw()` ; en plus, on peut décider leur couleur.

- Quelle fonction doit forcément être virtuelle ?

La fonction `draw` doit utiliser les fonctions de la librairie `CImg` pour dessiner la forme sur une fenêtre.

2.3 Q2 : Mémoriser les formes

Toutes les formes créés par l'utilisateur sont mémorisées dans un `vector`:

```
std::vector<Shape *> shapes;
```

```
Shape *p = new Rectangle(0,0, 100, 50);
shapes.push_back(p);
```

Pour dessiner toutes les formes, il faut juste appeler leur fonction `draw()` :

```
for (Shape *p : shapes) p->draw();
```

Écrire un programme où un ensemble de formes sont créés statiquement dans le main, et elles sont ensuite visualisées dans une fenêtre graphique avec `CImg`.

2.4 Q3 : Sélectionner une forme

Ajouter à la classe `Shape` la fonction

```
bool is_inside(int x, int y);
```

qui retourne `true` si le point (x, y) se trouve à l'intérieur de la forme.

- Est-ce que cette fonction doit être virtuelle ou pas ?

Ensuite, modifier le programme précédent pour donner la possibilité à un utilisateur de sélectionner et désélectionner une forme existante avec un clic de la souris à l'intérieur de la forme. Pour montrer que la forme est sélectionnée, on peut dessiner sa bordure avec une couleur différente. Si on clique à nouveau sur la forme, la forme sera désélectionnée.

Ajouter des méthodes `bool is_selected()` et `void select()` à la hiérarchie `Shape`. Est-ce que ces méthodes sont virtuelles ou pas ?

2.5 Q4 : Créer, déplacer, effacer des formes

Pour créer une forme, il faut :

- appuyer sur la touche 't' pour créer un triangle ;
- appuyer sur la touche 'r' pour créer un rectangle ;
- appuyer sur la touche 'c' pour créer un cercle.

Pour simplicité, les formes sont créées de la même taille, et sont toujours positionnées au milieu de la fenêtre.

Pour déplacer une forme, il faut d'abord la sélectionner, et ensuite appuyer sur la touche 'm'; le centre de la forme sera déplacé à la position où se trouve la souris.

On peut effacer une forme déjà sélectionnée avec la touche 'd'.

2.6 Q5 : Groupe

Il est possible de regrouper des formes pour créer une forme "composée". Pour faire ça, il faut créer une classe:

```
class ShapeGroup : public Shape {  
    vector<Shape *> shapes;  
    ...  
};
```

La classe contient une liste de formes, mais elle même est une forme. Pour dessiner un groupe, il faut juste dessiner toutes les formes contenues dans le groupe.

Pour créer un groupe, il faut

- sélectionner une forme;
- déplacer la souris sur une deuxième forme et appuyer sur la touche 'g';

- etcetera.

Quand on selectionne un groupe, toutes les formes sont dessinées avec une bordure colorée.

Pour decomposer un groupe, il faut le selectionner et appuyer sur la touche 'u'.

Pour deplacer ou pour effacer un groupe, on utilise la même methode que pour une forme simple. Quand on efface un groupe de formes, toutes les formes contenues dans le groupe sont effacées (les objets correspondants sont détruits).