VG101 — Introduction to Computer and Programming

Project 3

Manuel — UM-JI (Summer 2018)

- Include simple comments in the code
- If applicable, split the code over several functions
- Extensively test your code and impove it
- Write a single README file per assignment
- Zip all the files and upload the archive on Canvas

Project instructions

The goal of this project is to better understand Object Oriented Programming. In particular Classes, Inheritance, and Polymorphism are at the core of the project and must be applied in order to complete it. It is highly recommended to:

- Start thinking of the project as early as possible;
- Focus mainly on the organisation at the beginning;
- Define various objects and relate them to each others;

In this project many questions are left to the appreciation of the programmers. Based on your knowledge, research, and understanding argue on your choices in the README file.

The project splits into two parts: (i) the design of a generic parking lot, and (ii) the drawing of an interstellar parking lot using the OpenGL library. The two part should be written independently and be provided with their respective compiling commands.

A paper form will be provided for each student to briefly explain his contribution to the project.

Remark: do not exchange code among groups; Honor Code will be strictly applied.

Part I – A generic parking lot

In a software engineering company you are asked to design a software to manage a car park. Although several discussions with the customer have lead to some basic specifications much flexibility is left to you.

In order to show the customer how the program performs write a simulation where random vehicles enter and exit the parking lot over a given period of time.

Basic specifications

- Parking area: more than one floor, each one being of different size
- Vehicles: van, car, motorbike, bicycle
- Price: depends on the type of vehicle and time spent
- Arrival ticket: when a user arrives he receives a ticket containing:
  - Time of arrival
  - Type of the vehicle
  - Some information (hint) on where to find an empty slot
- Departure ticket: when a user leaves he receives a ticket containing:
  - Time spent in the parking lot
  - Type of vehicle
  - Price

Part II – An interstellar parking lot

The goal of this part is to use OpenGL to design an interstellar parking lot and drive a car into a free slot. The parking should be composed of at least ten slots, among which at least one is free. A slot that is not empty can be:

- Reserved for teleported vehicles. Such a slot contains a rectangle which randomly changes color;
- Occupied by a UFO which spins on itself;
- Occupied by a spacecraft which continuously zooms in and out;

The number of vehicles of each type as well as the amount of reserved slots is randomly set. In this initial setup the car is waiting in front of the barrier. Once open, the car follows a smooth trajectory to a free slot. The car should only stop *after* the empty space and reverse into the slot following a smooth curve. An example of such trajectories is drawn is blue and red on Fig. 1.

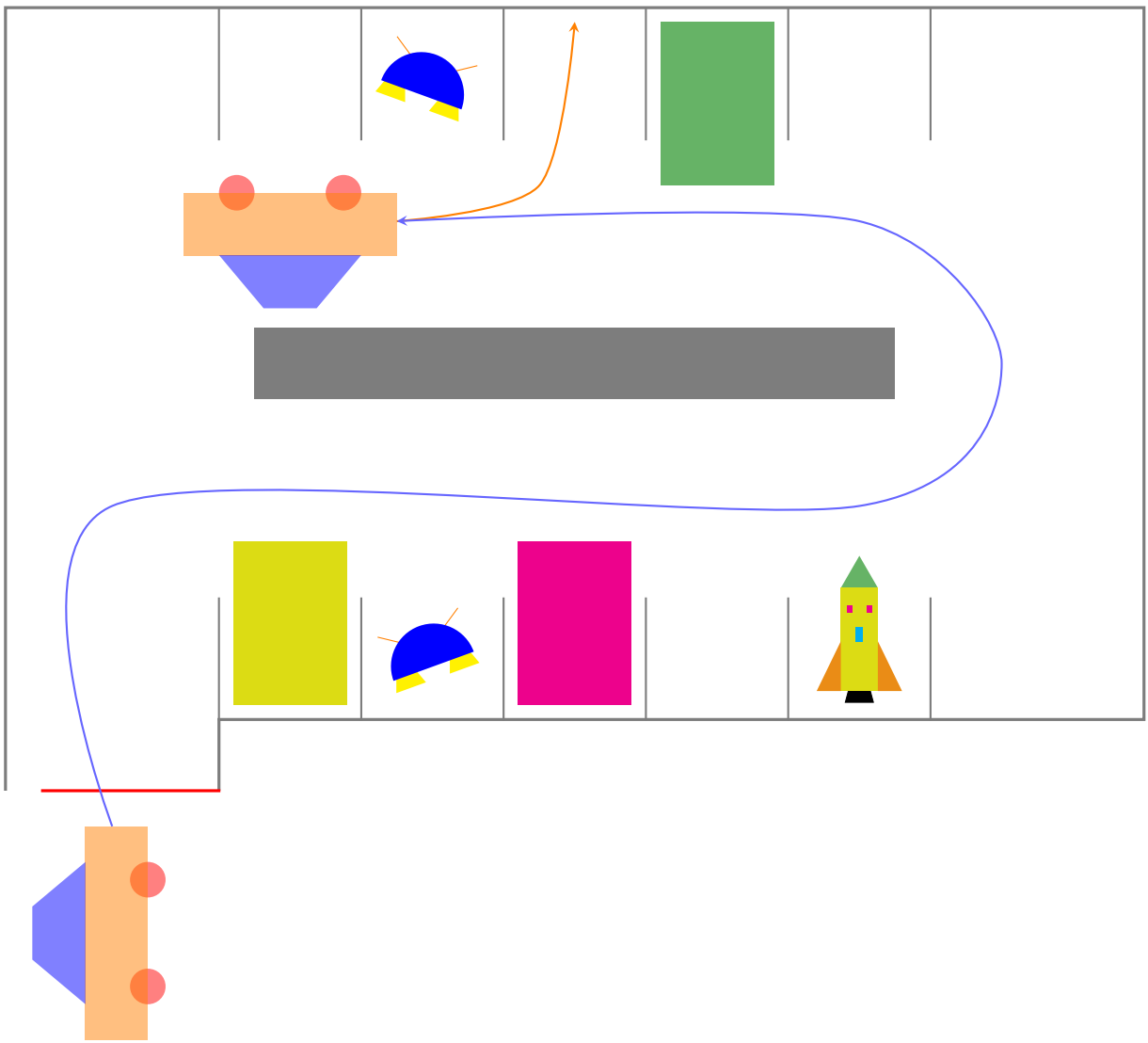

Figure 1: Interstellar car park: initial setup example

This part of the project being more complex it is advised the comply with the following guidelines.

- Define a clear hierarchy to organise the various objects (a partial version is provided on Fig. 2);
- Complete the partial classes interface provided below:
  - The `Vec` class defines a mathematical vector; The class should be immutable, i.e. no method is allowed to change any attribute at any time, but construction; It is intended for instance to define points without dealing with each coodinate;
  - The `figure` class defines a central point called `anchor` around which the figure can rotate, or zoom. Other methods are also listed;
  - The `Group` class inherits from `Figure` and as such is a figure. It is however composed of other "sub-figures", and as such can contain other `Group`;
- All the attributes should be either `private` or `protected`
- STL vectors can be used to store objects;

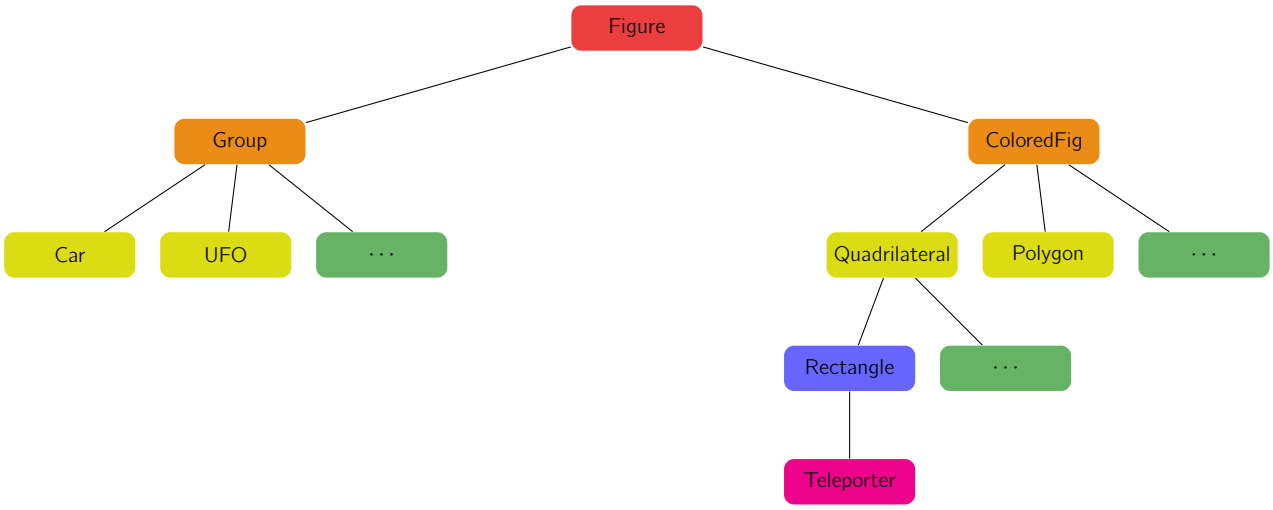

Figure 2: Partial interstellar parking slot hierarchy

Partial classes interface

```
class Vec {
private:
    float x,y;
public:
    Vec(float _x, float _y) {
        x = _x; y = _y;
    }
    float getX() {return x;}
    float getY() {return y;}

    // Example Overloads + operator
    // returns the sum of 2 Vec
    Vec operator+ (Vec v) {
        return Vec(x + v.getX(), y + v.getY());
    }
```

```
    // Overload - on 2 Vectors
    // return thier difference Vector

    // Overload * operator on a float k
    // return current Vector scaled by k

    // Overload * on 2 Vectors
    // return thier inner product (scaler product)

    // Overload << on an angle
    // return current vector rotated counter clockwise
    // by this angle

    // Overload >> on an angle
    // return current vector rotated clockwise
    // by this angle
};

class Figure {
protected:
    Vec anchor;
public:
    Figure() : anchor(0, 0) {}

    Vec getAnchor() {return anchor;}
    void setAnchor(Vec a) {anchor = a;}
    virtual void draw() = 0;
    virtual void move(Vec dir) = 0;
    virtual void rotate(float angle) = 0;
    virtual void zoom(float k) = 0;

    virtual ~Figure() {}
}

class Group : Figure {
private:
    // A Group of figure "has" other figures.
public:

    // We left out the constructor as well

    void draw();      // Draw out all its figures
    void move(Vec dir);    // Move all its figures
    void rotate(float angle);    // Rotate the group as a whole.
    void zoom(float k);     // Zoom the group as a whole.
    ~Group() {}
}
```

More advanced strategy

While applying the following advice is not mandatory, it can greatly help in the design of a clean project.

As the use of global variables is forbidden it is tempting to "abuse" static variables. However a cleaner way is to implement a `Singleton` class. A singleton is a clean way to ensure an object is not instantiated more than once. This could be useful for instance in the `glDraw` function.

More information on singleton can be found in the following resources:

- https://en.wikipedia.org/wiki/Singleton_pattern
- http://www.yolinux.com/TUTORIALS/C%2B%2BSingleton.html
- http://stackoverflow.com/questions/1008019/c-singleton-design-pattern

Minimal Singleton class implementation

```
class Singleton {
  private:
    Singleton() {};
    ~Singleton() {};
    // omit copy constructor
    // omit overloading assignment operator

Public:
    static Singleton* getInstance() {
      Static Singleton *s = NULL;
      if (s != NULL) s = new Singleton();
      return s;
    }

    Static void deleteInstance() {
      Singleton *s = Singleton::getInstance();
      if (s != NULL) delete s;
      s = NULL;
    }

};
```

Groups

Li Junhui, 李俊慧 (517370910070)
Hu Huinan, 胡慧楠 (517021910723)
Cheng Baisen, 程柏森 (517370910006)
Wang Hanyu, 王涵宇 (517370910174)

NICHOLAS CRAIG THOMAS ,715370290006)
Tai Chenxi, 邰晨希 (517021911049)
Zhang Dingkun, 张定坤 (517370910261)
Liu Zhendong, 刘镇东 (517370910053)

Wu Jiahao, 吴佳豪 (517370910203)
Yang Zhe, 杨哲 (517021911127)
Hu Honglei, 胡鸿磊 (517021910062)
Ni Yining, 倪毅宁 (515370910057)

Hou Shuyuan, 侯曙源 (517021910193)
Sun Zhuoyuan, 孙卓元 (517370910241)
Liu Yating, 刘雅婷 (517370910186)
Zhang Zhenyuan, 张震元 (517370910124)

Sun Yan, 孙彦 (517370910147)
Sheng Beiyi, 盛贝艺 (515370910110)
Wang Yuhao, 王宇昊 (517370910060)
Lu Hongru, 陆鸿儒 (517370910083)

ERVIN TJITRA (517370990027)
Yuan Yin, 袁寅 (517370910260)
Shen Shiyu, 沈诗语 (517370910132)
Zhang Yiwei, 张一苇 (517370910161)

Gu Zhenhao, 顾振昊 (517370910139)
Xu Jiawen, 许佳文 (517370910151)
Zhang Shutong, 张疏桐 (517370910180)
Shen Shiqi, 沈诗琪 (517370910100)

Shen Long, 申龙 (515370910037)
Liu Tianle, 刘天乐 (517370910223)
Jiang Yimin, 蒋逸民 (517370910111)
Xu Pengyu, 许鹏宇 (517021911183)

Liu Yongle, 刘永乐 (517370910212)
Hu Yifan, 胡艺帆 (517370910156)
Bai Ge, 白舸 (517370910105)
Shen Ting, 沈婷 (517370910004)

Zhou Yuankai, 周源锴 (517370910182)
Yang Tiancheng, 杨天成 (517370910259)
Luo Liangyu, 罗梁宇 (517370910113)
Xie Wenbo, 谢文博 (515370910183)

Wu Jiayao, 吴佳遥 (517370910257)
Chen Zhiyu, 陈祉余 (517370910210)
Wu Zheng, 吴争 (517370910176)
Hu Zhengdong, 胡正东 (517370910249)

Pan Zhihui, 潘智辉 (516370910174)
Yang Xuting, 杨煦庭 (517370910013)
Huang Mingjun, 黄铭珺 (517021911104)
Jiang Houfu, 姜厚甫 (517021911145)

Wang Yichao, 王逸超 (517370910011)
Chen Jiasheng, 陈家昇 (517021911034)
wang yixuan 汪怡暄 (517021910182)
Wang Yixuan, 王译萱 (517370910214)

Yuan Fangqi, 袁方祺 (517370910206)
Shi Li, 史历 (517370910032)
Shen Zhuowen, 沈卓文 (517370910057)
Tang Yiyang, 唐逸扬 (517370910041)

Wei Ye, 魏烨 (517370910175)
Liu Yatian, 刘亚天 (517370910010)
Ye Shunyi, 叶顺义 (517370910091)
ALEKSEI DANLI (517370990024)

Wang Xinyi, 王馨艺 (517370910188)
Huang Puyang, 黄浦阳 (517370910220)
s tan STANLEY HUAREN GU (517370990021)
Du Yang, 杜洋 (517370910195)

Chang Siyao, 常思尧 (517370910024)
Ji Wanying, 吉宛颖 (517370910185)
Liu Zhiyuan, 刘之远 (517370910240)
Wei Chenye, 危晨烨 (517370910150)

Zhao Zhijie 赵志劼 (517370910035)
He Yujia, 何雨佳 (517370910110)
JOHN MARK SERRA ,715370290019)
Bai Hongming, 白洪铭 (517370910192)

Li Hansen, 黎翰森 (517370910251)
Cai Zhenyi, 蔡臻一 (517370910216)
Chen Chulei, 陈初蕾 (517370910236)
Ye Roushuang, 叶柔霜 (516370910241)

Sun Qi, 孙琪 (517370910158)
MOHAMMADALI ASGARI VAZIRI (517370990025)
He Zhirui, 何智睿 (517021910117)
Feng Ruiquan, 冯睿泉 (517370910048)

zhao muhan 赵沐涵 (517370910215)
Ren Yi, 任易 (517370910115)
Jian Zeyu, 简泽宇 (517370910008)
Ye Siwei, 叶思威 (517370910122)

Song Yanbo, 宋彦伯 (517370910058)
li wen 李雯 (517370910099)
Jiang Yunpeng, 蒋云鹏 (517370910141)
Liu Zhejun, 刘哲竣 (516370910038)

Liu Yuelin, 刘越麟 (517370910252)
Chen Yaxin, 陈雅昕 (517021911020)
Zheng Jingnan, 郑静楠 (517370910103)
Jin Zhejian, 金哲健 (517370910167)

STEVEN WIJAYA (517370990038)
Zhang Kunhao, 张坤昊 (516370910232)
Chen Jiwen, 陈骥文 (517021911138)
Tim Shi, 时尖 (517370910255)

Liu Shuoyu, 刘朔钰 (517370910143)
Cao Jing, 曹静 (515370910055)
Chen Shuang, 陈爽 (517370910107)
Dong Chenyu, 董宸语 (517370910237)

Li Shiqi, 李诗琪 (517021910164)
Wu Yuhao, 吴宇豪 (517370910119)
DING ZHE GAN (517370990002)
Guan Kaiwen, 官凯文 (517021911143)

Lu Hengyi, 陆恒毅 (517370910054)
Zhang Xinli, 张新力 (517370910207)
Cheang Wenglam, 鄭穎琳 (517370910023)
AWAN OSAMA MALIK (516370990003)

Yang Tianqi 杨添淇 (517370910033)
Xu Zeheng, 许泽恒 (517370910205)
Zhao Ziming, 赵子铭 (517370910065)
Liu Zeyuan, 刘则远 (517370910029)

Zou Jiayun, 邹伽赟 (517370910191)
Chen Haoyang, 陈浩旸 (517370910046)
Liu Shiyi, 刘诗逸 (517370910157)
Chen Yibo, 陈逸波 (517370910138)

Zhao Yijia, 赵伊佳 (517370910243)
Sun Yi, 孙怡 (517370910187)
Kuai Yifeng, 蒯逸风 (517370910221)
Tan Yingwen, 谈颖雯 (517370910159)

Zhao Yiqing, 赵漪青 (517370910190)
Cai Tianzhang, 蔡天章 (517370910136)
wang tian yi 王天怡 (517370910160)
Xu Ang, 徐昂 (517370910090)

Zhuang Jiaying, 庄嘉滢 (517370910162)
Kuang Yue, 匡越 (517370910019)
Cai Weihao, 蔡炜灏 (517370910163)
Yang Haichao, 杨海潮 (517370910178)

Chen Yujie, 陈宇婕 (517370910183)
Chang Kenglun, 張耕綸 (517370910015)
Guo Yunjie, 郭蕴捷 (517370910166)