**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

BÁO CÁO BÀI TẬP LỚN

**Họ và tên sinh viên: Nguyễn Thị Kiều Chinh**

**Mã sinh viên: 21020752**

**Môn học: Lập trình nâng cao**

**Lớp học phần: INT2215 1**

# MỞ ĐẦU

## Lý do chọn dự án

Vốn là một con người yêu thích ngắm nhìn không gian vũ trụ vô tận, cùng với việc là một sinh viên khoa Công nghệ thông tin, tôi/em luôn mong muốn có thể sáng tạo ra một trò chơi để thỏa mãn niềm yêu thích đó của bản thân mình. Vì vậy, tôi/em chọn dự án game Comos có liên quan đến vũ trụ, một phần là cơ hội để bản thân tôi/em có cơ hội để tìm hiểu, học tập, ứng dụng những gì mình học được trong thực tế; mặt khác, đây cũng là dịp để tôi/em có thể tạo ra một game nho nhỏ của bản thân mình để giải trí sau những ngày mệt mỏi.

## Tổng quan về dự án

Dự án game của tôi/em có tên là Comos, trong tiếng anh có nghĩa là “vũ trụ”. Game được viết bằng ngôn ngữ [**C++**](https://vi.wikipedia.org/wiki/C%2B%2B) và sử dụng thư viện đồ họa [**SFML**](https://en.wikipedia.org/wiki/Simple_and_Fast_Multimedia_Library).

# NỘI DUNG

## Sơ bộ về ý tưởng và cách chơi

Luật chơi của game khá đơn giản. Khi bắt đầu màn chơi mới, người chơi sử dụng các phím lên, xuống, trái, phải trên bàn phím để di chuyển phi thuyền trên màn hình để tránh né các chướng ngoại vật như thiên thạch và tàu của người ngoài hành tinh. Ngoài ra, người chơi có thể nhấn phím A để bắn một thiên thạch bằng một viên đạn hoặc phím S để có thể hạ gục được nhiều mục tiêu trong phạm vi mà đạn có thể tiêu diệt.

* Về cách ghi điểm: Người chơi phải bắn trúng các thiên thạch và tàu của người ngoài hành tinh mới được tính điểm. Cách tính điểm cho từng loại được nêu trong game.
* Về cách thêm/mất mạng sống: Nếu người chơi điều khiển phi thuyền va vào thiên thạch hoặc tàu của người ngoài hành tinh hoặc bắn trúng vào các hành tinh thì sẽ bị mất một phần mạng sống. Ngược lại, nếu người chơi điều khiển phi thuyền đi đến các hành tinh thì sẽ được thêm một phần năng lượng(mạng sống).

Ngoài ra trong lúc chơi, người chơi có thể tạm dừng hoặc dừng lại.

## Tiến trình hoàn thành

Trước hết, em gói gọn cả game trong class gameEngine. Game gồm hai phần chính: phần mở đầu, menu là class Hello, phần game chủ yếu là class Game. Ngoài ra, còn có phần class GameOver có chức năng cho xem phần điểm sau mỗi lượt chơi. Em làm lần lượt phần class Game trước, sau đó đến phần class GameOver, cuối cùng mới làm menu ở class Hello.

### Phiên bản 1 – tạo class Game

#### Tạo các nhân vật cho game

|  |  |
| --- | --- |
| **Sự vật** | **Class** |
| Người chơi | Player |
| Thiên thạch và tàu vũ trụ | Object |
| Đạn | Bullet |
| Hành tinh | Planet |
| Nền | Background |
| Hình động nổ | Boom |
| Hình động trên nền | Star |

Còn phần điểm được lưu trong biến point, số lượng mạng còn lại trong health.

#### Thiết kế va chạm

* Giữa người chơi và màn hình: Khi vị trí của người chơi chạm đến cạnh của màn hình thì người chơi không di chuyển được nữa.
* Giữa người chơi và chướng ngoại vật: Khi người chơi tiếp xúc với chướng ngoại vật thì chướng ngoại vật đó sẽ bị xóa khỏi mảng lưu chướng ngoại vật đó.
* Giữa chướng ngoại vật với đạn: Nếu chướng ngoại vật tiếp xúc với đạn thì cả chướng ngoại vật và đạn đều bị biến mất. Lúc này chưa tính đến đoạn có khả năng tiêu diệt nhiều mục tiêu.
* Giữa các chướng ngoại vật, đạn với màn hình: Khi ra khỏi màn hình, các chướng ngoại vật sẽ bị xóa khỏi mảng đã lưu.

#### Nhận đầu vào

Game chủ yếu nhận đầu vào từ bàn phím để chơi. Lúc này chưa có chuột vì chưa cần tạo menu và lựa chọn.

### Phiên bản 2 – tạo class GamOver

Trong phần này, chủ yếu sử dụng phần Text và Texture để tạo các hình ảnh, chữ viết trên màn hình và sử dụng đầu vào là chuột để lựa chọn. Phần này khá đơn giản, và lúc đó em vẫn chưa có các chức năng chơi lại.

Ngoài ra trong thời gian này, em đã đầu làm phần class Hello với việc render một số lựa chọn nhất định.

### Phiên bản 3 – tạo class Hello hoàn chỉnh

Em đã đưa các lựa chọn vào các vòng loop để có thể lựa chọn hoặc quay lại, không chỉ riêng phần class Hello và class GameOver. Ngoài ra, em bắt đầu thêm tính năng bắn một viên đạn nhưng có thể tiêu diệt được các chướng ngoại vật xung quanh nó.

### Phiên bản 4 – Chèn thêm âm thanh

### Phiên bản 5 – Thêm chế độ tạm dừng/tiếp tục/dừng hẳn

## Đánh giá của bản thân

* Đây là một game khá là đơn giản, không có quá nhiều thuật toán hay logic game phức tạp, chủ yếu dựa trên tầm hiểu biết và ý tưởng sáng tạo của riêng mình/em.
* Game có các chức năng khá đầy đủ như có menu, có chọn nhân vật chơi, có hình ảnh, âm thanh, chế độ tạm dừng/tiếp tục,… và có khả năng phát triển hơn nữa như có thể lưu trữ các lượt chơi của từng người, cho xem được những người chơi có số điểm cao nhất,…
* Về đồ họa và âm thanh của game còn khá sơ sài, chủ yếu là do mình/em tìm kiếm trên mạng internet và chỉnh sửa để phù hợp với game một cách tốt nhất.
* Phần báo cáo do thời lượng cũng như giới hạn độ dài không thể nói rõ các chi tiết được, đặc biệt trong phiên bản 1, em chia làm nhiều quá trình làm xuyên suốt, nên phần báo cáo sẽ có chỗ chưa đầy đủ, mong thầy xem xét.

## Tài liệu tham khảo và nguồn đồ họa

* Game chủ yếu là do mình/em tự làm, có tham khảo phần Thiết kế va chạm ở [đây](https://github.com/Headturna/SFML_C-_Simple_Games/tree/master/SFML_C%2B%2B_Simple_Games/SFML_C%2B%2B_Simple_Games/Game%203). Ngoài ra, mình/em còn tự học về SFML ở [đây](https://www.sfml-dev.org/tutorials/2.5/).
* Nguồn hình ảnh chủ yếu mình/em lấy trên [Pinterest](https://www.pinterest.com/), không biết có permission của tác giả hay không nhưng mình/em làm là phi lợi nhuận và không public ra bên ngoài.
* Nguồn âm thanh chủ yếu mình/em tìm Sound Effects trên [Youtube](https://www.youtube.com/).