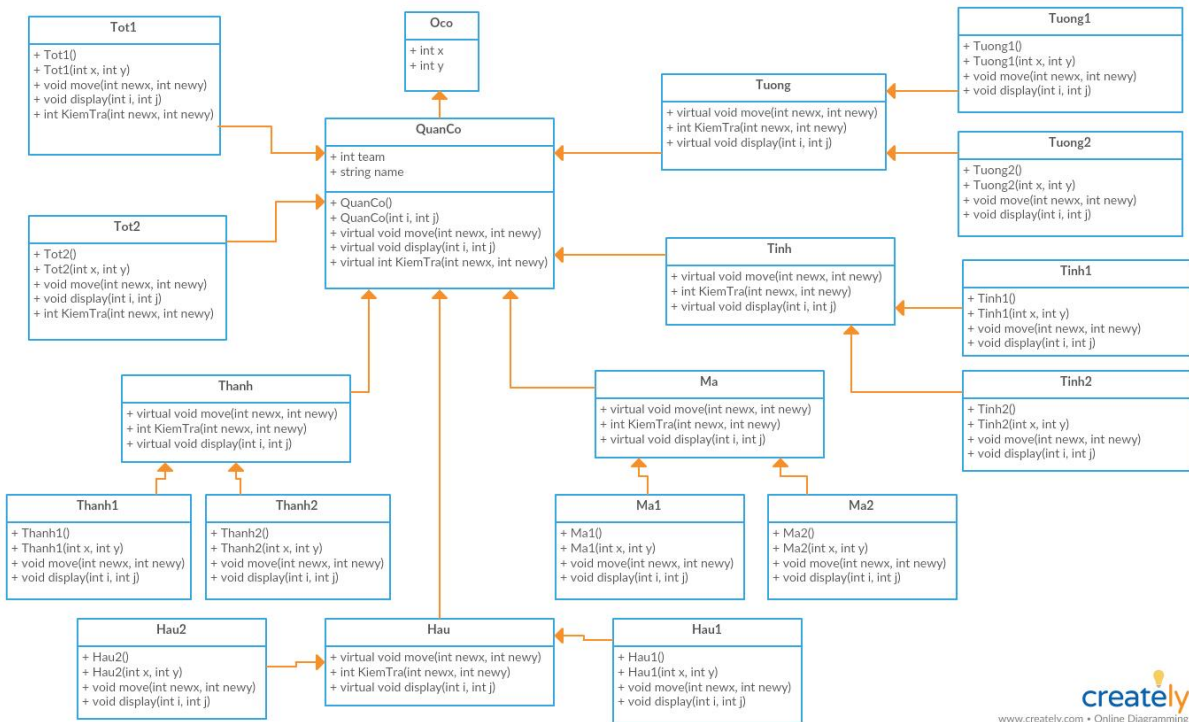


Giải thích các lớp và các hàm



Sơ đồ lớp

I. Các lớp

1. Class Oco

+Thuộc tính:

Int x, int y : Tọa độ ô cờ

2. Class QuanCo (kế thừa Oco) // 64 ô cờ ⇔ 64 QuanCo

+Thuộc tính:

Int team -> quyết định xem ô cờ rỗng, team 1, team 2

String name -> Cho biết chính xác tên của quân cờ (null là quân cờ rỗng).

+Phương thức:

3 hàm ảo:

virtual void move(int newx, int newy) -> di chuyển đối tượng đến tọa độ newx, newy

virtual void display(int i, int j) -> Vẽ quân cờ lên bàn cờ tại vị trí i, j.

virtual int KiemTra(int newx, int newy) -> Kiểm tra vị trí newx, newy có hợp lệ không

3. Class Tot1 (kế thừa QuanCo) // Lớp tốt team 1

-> Cài lại 3 hàm :

void move(int newx, int newy)

void display(int i, int j

int KiemTra(int newx, int newy

4. Class Tot2 (kế thừa QuanCo) // Lớp tốt team 2

-> Cài lại 3 hàm :

void move(int newx, int newy)

void display(int i, int j

int KiemTra(int newx, int newy

5. Class Tuong (kế thừa QuanCo) // Lớp tướng

-> Cài lại hàm int KiemTra(int newx, int newy)

6. Class Tuong1 (kế thừa Tuong) // Lớp tướng team 1

-> Cài lại 2 hàm:

void move(int newx, int newy)

void display(int i, int j)

7. Class Tuong2 (kế thừa Tuong) // Lớp tướng team 2

-> Cài lại 2 hàm:

void move(int newx, int newy)

void display(int i, int j)

8. Class Tinh (kế thừa QuanCo) // Lớp tinh

-> Cài lại hàm int KiemTra(int newx, int newy)

9. Class Tinh1 (kế thừa Tinh) // Lớp tinh team 1

-> Cài lại 2 hàm:

void move(int newx, int newy)

void display(int i, int j)

10. Class Tinh2 (kế thừa Tinh) // Lớp tinh team 2

-> Cài lại 2 hàm:

void move(int newx, int newy)

11. Class Thanh (kế thừa QuanCo) // Lớp thành

-> Cài lại hàm int KiemTra(int newx, int newy)

12. Class Thanh1 (kế thừa Thanh) // Lớp thành team 1

-> Cài lại 2 hàm:

void move(int newx, int newy)

void display(int i, int j)

13. Class Thanh2 (kế thừa Thanh) // Lớp thành team 2

-> Cài lại 2 hàm:

```
void move(int newx, int newy)
```

14. Class Ma (kế thừa QuanCo) // Lớp mã

-> Cài lại hàm int KiemTra(int newx, int newy)

15. Class Ma1 (kế thừa Ma) // Lớp mã team 1

-> Cài lại 2 hàm:

```
void move(int newx, int newy)
```

```
void display(int i, int j)
```

16. Class Ma2 (kế thừa Ma) // Lớp mã team 2

-> Cài lại 2 hàm:

```
void move(int newx, int newy)
```

17. Class Hau (kế thừa QuanCo) // Lớp hậu

-> Cài lại hàm int KiemTra(int newx, int newy)

18. Class Hau1 (kế thừa Hau) // Lớp hậu team 1

-> Cài lại 2 hàm:

```
void move(int newx, int newy)
```

```
void display(int i, int j)
```

19. Class Hau2 (kế thừa Hau) // Lớp hậu team 2

-> Cài lại 2 hàm:

```
void move(int newx, int newy)
```

II. Các hàm

+ Các hàm tham khảo từ Internet:

```
// Hàm xóa màn hình.
```

```
void XoaManHinh();
```

```
// Hàm dịch chuyển con trỏ đến tọa độ x, y.
```

```
void gotoxy(int x, int y);
```

```
// Hàm tô màu.
```

```
void textcolor(int x);
```

```
// Chỉnh màn hình CMD
```

```
void SetWindow(int Width, int Height);
```

```
// Thời gian thực
```

```
const std::string currentDate();
```

```
// Hàm ẩn đi con trỏ
```

```
void Nocursortype();
```

```
// Hàm chơi file wav
```

```
sndPlaySound();
```

+ Các hàm tự lên ý tưởng:

```
// Hàm vẽ ô cờ tại tọa độ x, y, với tham số t -> quyết định màu sắc
```

```
void VeOCo(int t, int x, int y);
```

```
// Hàm vẽ bàn cờ
```

```
void VeBanCo();
```

```
// Hàm làm sáng ô cờ tại tọa độ i, j, với color
```

```
void LamSang(int i, int j, int color);
```

```
// Hàm hủy sáng ô cờ tại tọa độ i, j.
```

```
void HuySang(int i, int j);
```

```
// Xuất lượt đi lên màn hình
```

```
void XuatLuotDi(int team);
```

```
// Hàm xác định quân cờ ( với nút nhập vào là c, tọa độ đang chọn là x, y)-> kết thúc hàm thì x, y là quân cờ được chọn.
```

```
void XacDinhQuanCo(char c, int &x, int &y);
```

```
// Các hàm in ra Menu
```

```
void InMenu();
```

```
void Inp1vsp2();
```

```
void Inhistory();
```

```
void Inexit();
```

```
void Inlaw();
```

```
// Các hàm in thông báo
```

```
void QuanCoKhongHopLe(int team) -> Thông báo cho team biết quân cờ chọn vào không hợp lệ
```

```
void HuyNuocDi(int team)-> Hủy quân cờ đã chọn -> chọn lại quân cờ khác
```

```
void ViTriKhongHopLe(int team) -> Thông báo vị trí di chuyển tới không hợp lệ
```

```
void ThongBaoNguoiWin(int team) -> Thông báo kết quả ván đấu với team =0/hòa, 1// team 1 win/ 2// team 2 win/
```

```
// Lịch sử chơi
```

```

void GhiVaoLichSu(char* ten1, char* ten2, int KetQua) -> ghi vào lịch sử tên 1, tên2, kết quả

// Cầu hòa

int CauHoa(int team)- > team muốn hòa

int DongYHoa(int team) -> team đồng ý hòa

// Chơi nhạc

void PlayMusic(int n); -> 3 chế độ nhạc

void StopMusic() -> tắt nhạc

void TaoBanCo() - > tạo ra bàn cờ 8x8

void XuatBanCo(char* ten1, char* ten2) -> Xuất ra bàn cờ và tên 2 người chơi

int CoNuocDi(QuanCo* &x) -> 1: quân cờ có nước đi/ 0: quân cờ không thể di chuyển

void InCacViTriKhaThi(int x, int y) -> Khi người chơi chọn vào 1 quân cờ thì in ra các vị trí quân cờ đó có thể di chuyển tới/

void XacDinhViTri(char c, int x, int y, int &newx, int &newy)-> Xác định vị trí di chuyển tới

void PhongTot(int newx, int newy, int team); -> Cơ chế phong cấp cho tốt

void Lawofchess() -> In ra luật cơ bản của các quân cờ

void Intro(char* ten1, char* ten2) -> Nhập vào tên 2 người chơi

void ThongBaoQuanMat(int ntot, int nthan, int nma, int ntinh, int nhau, int team)-> Xuất ra màn hình số quân mất của mỗi bên.

void ThongBaoChieuTuong(int team)- > thông báo cho team biết là đang bị chiếu

int MuonDauHang(int team) -> team muốn đầu hàng

int NhanVaoNutDauHang(char c, int team, char* ten1, char* ten2)- > xem có nhấn vào nút đầu hàng không

int NhanVaoNutCauHoa(char c, int team, char* ten1, char* ten2) -> xem có nhấn vào nút cầu hòa không.

void ChoiNhac(int n, int m) : n: on/off. m: chế độ chơi

void Sound(int &music, int &n, char c);

void XuLiGame();

void GiaiPhong();

```

