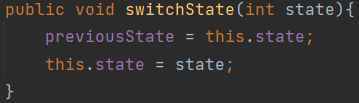
METHOD

Là phương thức được gọi đến trong method Update() với case TUTORIAL. Khi được gọi đến, nó sẽ update lại các thông số của game, nhân vật khi game đến thời điểm quan trọng như intro đầu game, gặp boss

1. switchState(int state)



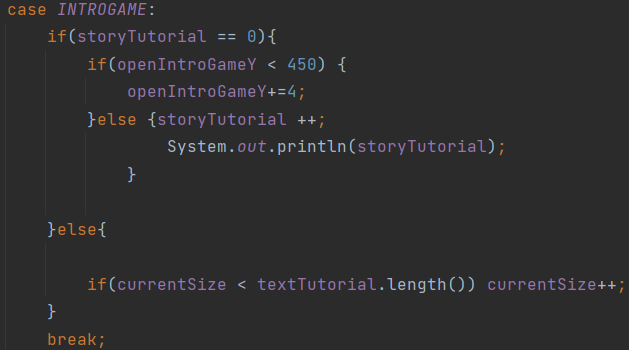
Là hàm chuyển đổi trạng thái của game, tham số truyền vào chính là trạng thái mới của game. Còn tham số hiện tại (this.state sẽ được gán vào biến previousState là trạng thái trước đó)

2. TutoriralUpdate()

Đây là method update lại các thông số



Với tham số truyền vào mệnh đề switch là tutorialState tức là các giai đoạn quan trọng của game như khởi đầu game, gặp boss hay chiến thắng, cụ thể thì tutorialSate có các giá trị là INTROGAME và MEETFINALBOSS.

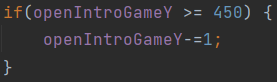


Đối với case INTROGAME, khi biến storyTutorial bằng 0, nếu openIntroGameY < 450 thì openIntroGameY += 4. Ngay khi vào game thì ta có thể thấy một màn đen từ từ mở ra, và tốc độ mở ra nhanh hay chậm phụ thuộc vào giá trị mà openIntroGameY cộng thêm. Do hàm Update chứa method TutorialUpdate() chứa trong vòng lặp while của class GamePanel() nên case INTROGAME sẽ chạy cả nhánh else do openIntroGameY sẽ > 450. Khi đó biến storyTutorial tăng lên 1 đv.

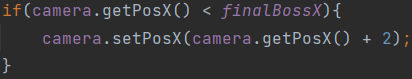
Khi biến storyTutorial đã được tăng lên 1 đv, lúc này nó sẽ khác 0, nhánh else bên dưới sẽ được thực thi. Mục đích của việc tăng biến currentSize để mỗi lần Update xong thì trong Render nó sẽ in lần lượt các đoạn text ở đầu game



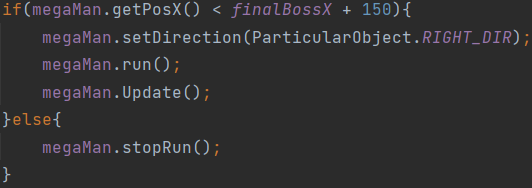
Đối với case MEETFINALBOSS, trước tiên chúng ta cũng kiểm tra xem biến storyTutorial có bằng 0 hay không.



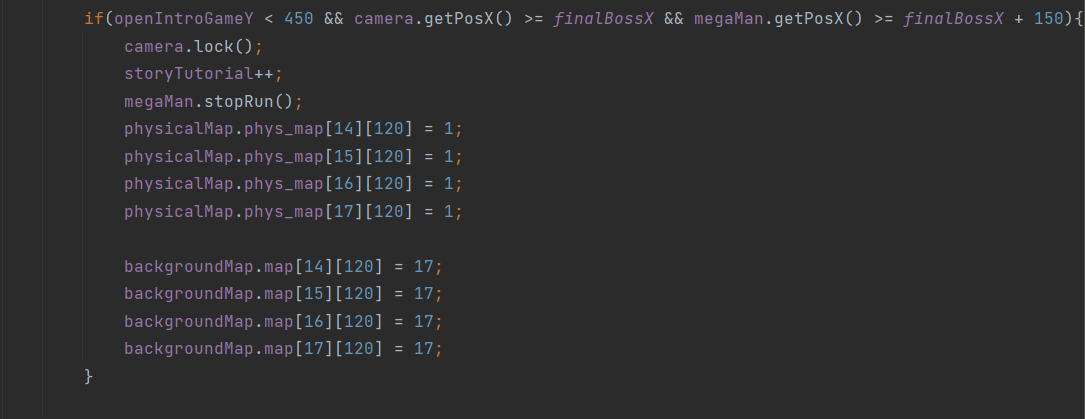
Nếu đúng ta kiểm tra tiếp xem biến openIntroGameY có ≥ 450 hay không, rõ ràng trước đó biến openIntroGameY đã lớn hơn 450 do ta đã cộng thêm trong case INTROGAME, việc giảm openIntroGameY đơn giản là kéo viền đen ở dưới và trên màn hình về giữa sau mỗi lần update, ngược lại với case INTROGAME, tốc độ nhanh chậm phụ thuộc vào giá trị trừ đi lớn hay bé.



Lệnh kiểm tra này để đưa vị trí camera hiện tại khi nhân vật gặp FinalBoss đến gần vị trí boss hơn nếu vị trí camera hiện tại nhỏ hơn finalBossX tức là khi nhân vật tiến gần vào khu có Boss.



Phía trên chúng ta đã dịch camera về gần phía Boss, lệnh if-else này sẽ đưa nhân vật tiến về gần Boss hơn. Cụ thể nếu vị trí nhân vật đang nhỏ hơn vị trí boss + 150 pixel, ta set hướng nhân vật về bên phải do boss bên phải, cho nhân vật chạy, update lại trạng thái cho nhân vật đến khi nào đến vị trí thoả mãn điều kiện thì dừng chạy.

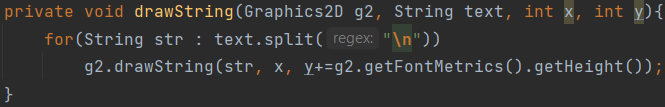


Tổng hợp các điều kiện phía trên là camera, nhân vật đến vị trí yêu cầu, viền đen đã hạ xuống thì lúc này camera sẽ bị khoá do khu giao tranh nằm trọn trong khung hình, nhân vật ngừng chạy và cánh cửa hang boss cũng sẽ bị thay thế thành vật cản không thể ra vào.



Khi biến storyTutorial được tăng lên, GamePanel sẽ chạy tiếp method Render, sau khi Render xong lại tiếp tục Update, lúc này storyTutorial đã lớn hơn 0, hàm TutorialUpadte() với case MEETFINALBOSS sẽ chạy nhánh else để update currentSize. Biến currentSize trong TutorialRender sẽ giúp ta in ra đoạn thoại của các nhân vật (hiện chưa có thoại giữa hai nhân vật Megaman và FinalBoss)

3. drawString(Graphics2D g2, String text, int x, int y)



Tham số truyền vào là lớp đồ hoạ Graphics, đoạn text và toạ độ vị trí đoạn text. Với mỗi đoạn text con được ngăn cách bởi ký tự xuống dòng, ta tiến hành vẽ text. Toạ độ y có chút thay đổi do cộng thêm với chiều cao của font chữ bằng phương thức g2.getFontMetrics().getHeight()

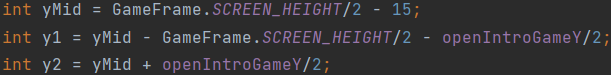
4. TutorialRender(Graphics2D g2)



Tương tự như TutorialUpdate, method TutorialRender cũng phải chia thành các case như nhau.



Đối với case INTROGAME, ta cần tạo thêm một số biến để đơn giản hoá việc vẽ nền



Biến yMid ở đây là toạ độ y giữa màn hình lệch lên trên một chút. Biến y1 thông qua yMid, ½ chiều cao màn hình game và ½ toạ độ openIntroGameY tăng dần sau mỗi lần game Update chính là toạ độ y của lớp nền đen phía trên khi vào INTROGAME, còn y2 là toạ độ y của lớp nền đen dưới.

y1

Ta có thể hiểu sơ qua như hình dưới

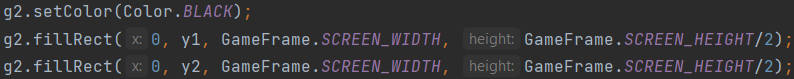
y1

y1

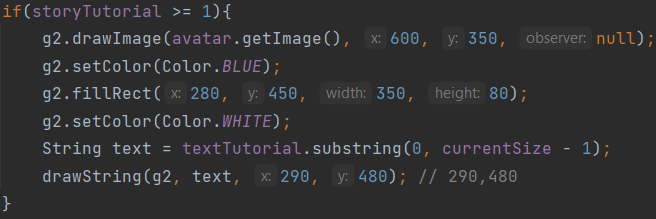
y2

y2

y2

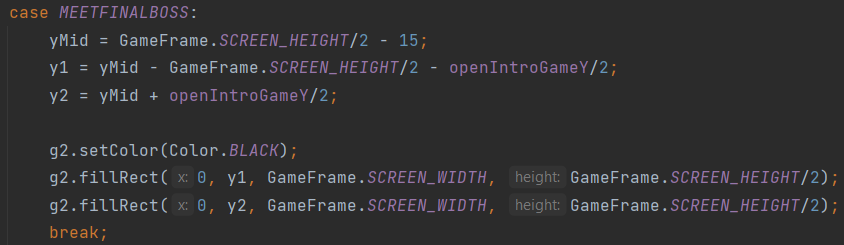


Sau khi đã có thông số y1, y2 ta sẽ vẽ những hình chữ nhật đen theo toạ độ chung x = 0, độ rộng bằng độ rộng màn hình game, chiều cao bằng ½ chiều cao màn hình game



Sau khi hai viền đen mở ra, lúc này biến storyTutorial đã được cộng thêm trên method TutorialUpdate, method TutorialRender sẽ vẽ ra hình như dưới cho đến khi hết thoại





Đối với case MEETFINALBOSS, hàm sẽ vẽ thêm hai viên đen như hình dưới

