

برنامه مربی - آموزش شماره ۱

تعیین نوع بازیکنان توسط عامل مربی

یکی از کارهایی که عامل مربی می تواند انجام دهد این است که نوع بازیکنان را تعیین کند .

در سرور ۱۳ ، ۱۸ نوع برای بازیکن وجود دارد و هر نوع دارای خصوصیات متفاوتی می باشد . لیست زیر خصوصیات است که هر بازیکنی آن ها را داراست :

dPlayerSpeedMax

dStaminaMax

dPlayerDecay

dInertiaMoment

dDashPowerRate

dPlayerSize

dKickableMargin

dKickRand

dExtraStamina

dEffortMax

dEffortMin

dMaximalKickDist

در هر بار اجرای سرور ، بازیکنان از طرف سرور خصوصیات با مقادیر جدیدی ، به طور تصادفی دریافت می کنند . به علت اینکه نوع ها در هر بار تصادفی می آیند هیچ نوعی (مثلا نوع ۱) را نمی توان یافت که هر بار با خصوصیات یکسانی ایجاد شود . برای مثال شاید یک بار نوع شماره ۱ دارای kickableMargin زیادتر از حد معمول و بیشتر از دیگران باشد و بار دیگر کمترین kickableMargin را داشته باشد !

یک محدودیت هم وجود دارد که عامل مربی از هر نوع فقط می تواند برای یک بازیکن استفاده کند . یعنی مثلا دو بازیکن ۷ و ۱۱ نمی توانند نوع یکسانی مثلا ۲ داشته باشند .

حال تنها کاری که ما باید بکنیم همین است ! بازیکنان را براساس ویژگی هایشان مرتب کنید ! برای مثال شما می خواهید بازیکن شماره ۹ ، چاق ترین (بیشترین kickableMargin) بازیکن شما باشد . برای این منظور نمی توانید بگویید مثلا ۹ باید همیشه

نوع ۵ باشد ! بلکه باید یک تابع بنویسید که بازیکنان را براساس kickableMargin مرتب کند و سپس نوعی که دارای kickableMargin بیشتری نسبت به بقیه هست را به بازیکن شماره ۹ دهد . کاری که خود بیس UVA trilearn در ابتدا انجام داده همان کاری است که باعث شده خصوصیات بازیکنان تصادفی بیاید . یعنی به هر بازیکن یک مقدار مشخص داده به صورت زیر و همین باعث شده هر بار بازیکنان با خصوصیات تصادفی مختلفی به زمین بیایند !!

```
substitutePlayer(2,1);  
substitutePlayer(3,2);  
substitutePlayer(4,3);  
substitutePlayer(5,4);  
substitutePlayer(6,5);  
substitutePlayer(7,6);  
substitutePlayer(8,7);  
substitutePlayer(9,8);  
substitutePlayer(10,9);  
substitutePlayer(11,10);
```

در حالی که نوع بازیکن باید توسط تابعی که مرتب سازی انجام می دهد مشخص شود . مثال :

```
substitutePlayer(2,KickableMarginMax());
```

که در مثال بالا تابع kickableMarginMax شماره نوعی که بیشترین KickableMargin را دارد مشخص می کند .

در بیس UVA trilearn برای اداره ی مربی باید به فایل BasicCoach.cpp مراجعه کنید .

برای اینکه بخواهید پی ببرید مثلا kickableMargin نوع شماره یک چند است باید از دستور زیر استفاده کنید :

```
WM->getInfoHeteroPlayer(1).dKickableMargin
```

و البته به جای dKickableMargin می توانید یکی دیگر از خصوصیات که در ابتدای مقاله قرار گرفت را استفاده کنید و مقدار آن را بدست آورید . و همچنین به جای عدد یک می توانید یک عدد دیگر (یک نوع دیگر - بین ۰ تا ۱۷) را انتخاب کنید .

خوب آموزش این قسمت تموم شد . شما می توانید از لینک زیر BasicCoach.cpp و BasicCoach.h را که من آپلود کردم دانلود کنید و نحوه مرتب سازی را مشاهده کنید ! من در آخر فایل BasicCoach.cpp یک تابع اضافه کردم که بازیکن ها

رو براساس kickableMargin مرتب می کنه ! شما می توانید به جای kickableMagin بر اساس ویژگی های دیگر مرتب کنید (خصوصیتی که در ابتدای مقاله گفتم) . البته لیست این ویژگی ها در انتهای فایل ServerSetting.h نیز موجود است .

نکته مهم : ویژگی های dPlayerSpeedMax و dPlayerSize در همه انواع همیشه مقادیر ثابتی هست و تصادفی نمی آیند و مقادیر همه انواع هم با یکدیگر در این دو ویژگی برابرند ! پس حواستان باشد که بازیکنان را براساس این ویژگی ها مرتب نکنید ...

دانلود فایل ها :

<http://robotoos.persiangig.com/UVA%20trilearn/Coach/BasicCoach.tar.gz>

یا

<http://robotoos.persiangig.com/UVA%20trilearn/Coach/BasicCoach.zip>

در صورت خرابی لینک به انجمن های www.robotoos.com مراجعه کنید .

تپیه شده توسط تیم روباتوس : شایان صالحیان و سعید عسکری

تماس با ما :

www.robotoos.com

Robotoos2D@gmail.com