

الگوریتم تشخیص برخورد (Collision Detection)

در بازی شما، اشیایی وجود دارد. این اشیاء ممکن است به هم برخورد کنند. اگر آنها برخورد کردند، شما ممکن است بخواهید یکی یا هر دوی آنها را از بین ببرید یا ممکن است بخواهید که آنها پخش کنند. در هر صورت شما باید توانایی تشخیص رویداد برخورد را داشته باشید. همچنین ممکن است شما لازم باشید بدانید که میزان برخورد چقدر بوده است. در ادامه، برای سوالهای **اگر و چقدر** برخورد اشیاء شما، باید راهی برای بهینه سازی آن وجود داشته باشد و راهی که به اندازه کافی برای بازی کننده های بازی شما دقیق باشد.

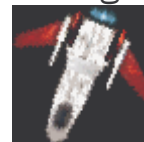
من نمی خواهم به شما چگونگی بهینه سازی تشخیص برخورد را نشان دهم بلکه می خواهم به شما نشان دهم که چگونه این کار را انجام می دهند، Pixel-Perfect. هر بهینه سازی تا حد ممکن به عهده خود شما است.

با معرفی دو شی زیر شروع می کنیم:

شی A



شی B



گوشه های بالا چپ این اشیاء مختصاتی دارند که ما بایستی آنها را $(AX1, AY1)$ و $(BX1, BY1)$ بنامیم. گوشه های پایین راست این اشیاء مختصاتی دارند که ما بایستی آنها را $(AX2, AY2)$ و $(BX2, BY2)$ بنامیم. در اینجا فقط یک راه وجود دارد که برخورد رخ دهد و آن وقتی است که دو مستطیل $(AX1, AY1)-(AX2, AY2)$ و $(BX1, BY1)-(BX2, BY2)$ رویهم افتاده باشند.

ما می توانیم بعضی از شروط را که در آنها امکان برخورد وجود ندارد حذف کنیم.

چون $AX1 < AX2$ در همه حالتها درست است (و همینطور برای $AY1 < AY2$ و $BX1 < BX2$ و $BY1 < BY2$) ما می توانیم شروط زیر را رد کنیم:

$BY2 < AY1$
 $AY2 < BY1$
 $BX2 < AX1$
 $AX2 < BX1$

اگر هر کدام از روابط بالا True باشد، سپس هیچ برخوردی نمی تواند اتفاق افتاده باشد. اگر همه روابط بالا False باشند، سپس یک برخورد می تواند اتفاق افتاده باشد. اگر ما نتوانیم یک برخورد را بطور قطعی تشخیص دهیم باید بررسی بیشتری انجام دهیم.

حالا ما می خواهیم پیدا کنیم که چقدر این دو مستطیل رویهم قرار گرفته اند. ما مکان این رویهم رفتگی را در $(CX1, CY1)-(CX2, CY2)$ تعیین می کنیم.

اگر $AX1 < BX1$ باشد آنگاه $CX1 = BX1$ و $CX2 = AX2$ است در غیر اینصورت $CX1 = AX1$ و $CX2 = BX2$ است. اگر $AY1 < BY1$ آنگاه $CY1 = BY1$ و $CY2 = AY2$ است در غیر اینصورت $CY1 = AY1$ و $CY2 = BY2$ است.

اجازه دهید نگاه بیشتری در مثال تصویری آن بیندازیم:



یک حالت از برخورد

حالا اجازه دهید اشیا را با مستطیل‌های محیطی آنها جایگزین کنیم.



شی A اکنون با مستطیل آبی جایگزین شده است و شی B با مستطیل قرمز. مستطیل ارغوانی نیز قسمت روبه‌هم رفتگی را نشان می‌دهد.

داخل مستطیل ارغوانی مانند شکل زیر است:



این نکته مهم است که باید توجه کنید که فقط این ناحیه بسیار کوچک است که شما برای تشخیص برخورد بررسی می‌کنید. بنابراین، اجازه دهید به قسمتهایی از هر کدام از اشیا که باعث ساخت این تصویر شده است نگاهی بندازیم:

شی A:



شی B:



اگر ما این مستطیل را پیکسل به پیکسل پیمایش کنیم، جایی را خواهیم دید که در آن هر دو تصویر دارای یک پیکسل هستند که رنگ آن سیاه (یا هر رنگی که به عنوان رنگ شفاف تعیین شده است) نیست، ما به نتیجه زیر خواهیم رسید:



تصویر حاصل وقتی که هیچ پیکسلی ست نشده باشد سیاه، هر پیکسل ست شده سفید، و جایی که هر دو ست شده باشند قرمز است. تنها یک پیکسل قرمز دلالت بر وقوع یک برخورد دارد. شما می توانید در صورت اهمیت مقدار برخورد، تعداد پیکسلهای قرمز را بشمرید. در صورتی که فقط بررسی پیکسلهای روبه‌هم رفته مورد نظر باشد به محض یافتن یک پیکسل می توانید بگویید که یک برخورد اتفاق افتاده است.

امیدوارم مطالعه این مقاله یک دید کلی نسبت به الگوریتمهای تشخیص برخورد به شما داده باشد.

نویسنده: **TANSTA AFL** (tanstaaf@gamedev.net)

مترجم: محمد صافدل

منبع: <http://www.GameDev.net>