

Boombberman



آشنایی:

Bomberman یکی از بازی‌های محبوب کنسول بازی Nintendo است که در دهه هفتاد میلادی برای اولین بار معرفی شد. اولین نسخه از این بازی به صورت تک‌نفره عرضه شد و در آن بازیکن باید با از بین بردن دشمن‌ها و پیدا کردن درب مرحله بعد، مراحل را سپری می‌کرد و در مرحله‌های بعد با چالش‌های سخت‌تر و هیجان انگیزتر برخورد می‌کرد.

از طریق لینک زیر می‌توانید نمونه‌ای شبیه سازی شده از نسخه قدیمی این بازی را ببینید و بازی کنید:

http://www.retrogames.cz/play_085-NES.php?language=EN

معرفی بازی:

در نسخه اولیه این بازی یک بازیکن با شروع از اولین مرحله، در هر مرحله باید درب خروج از مرحله و رسیدن به مرحله بعد را پیدا کند و پس از از بین بردن دشمن‌های خود در این مرحله با قرار گرفتن بر روی این درب به مرحله بعد برسد.

در هر مرحله بازیکن در یک نقشه مشابه الگوی شکل 1 قرار دارد. نقشه یک مرحله از تعدادی بلوک مربعی شکل تشکیل شده است که به صورت یک شبکه مستطیلی (مثلا 80 در 20) یکدیگر قرار گرفته‌اند. در بلوک‌هایی از نقشه که طول و عرض مختصات آن هر دو زوج هستند بلوک‌ها به صورت سنگ هستند. سایر بلوک‌های نقشه می‌تواند خالی یا دارای دیوار باشد.

شکل 1: شکل کلی نقشه یک مرحله از بازی

بازیکن در ابتدا می‌تواند در هر گام در بلوک‌های خالی اطراف خود جا به جا شود. به علاوه می‌تواند در بلوکی که در حال حاضر در آن قرار دارد بمب بگذارد. یک بمب پس از 5 ثانیه منفجر می‌شود و تمام دیوارها و دشمن‌ها و خود بازیکن را در صورتی که در فاصله افقی یا عمودی 1 از محل انفجار باشند از بین می‌برد. به علاوه در صورتی که در زمان انفجار یک بمب، بمب دیگری در محدوده انفجار آن باشد، این بمب نیز منفجر خواهد شد.

در ابتدا بازیکن تنها می‌تواند یک بمب در هر لحظه در صفحه داشته باشد. اما این ویژگی‌ها ممکن است با استفاده بازیکن از قوی‌کننده یا ضعیف‌کننده تغییر کند که در ادامه آنها را معرفی می‌کنیم.

در هر مرحله تعدادی دشمن وجود دارد که در مرحله این دشمن‌ها متنوع‌تر و با ویژگی‌های جدید می‌شوند. برخی از این دشمن‌ها را در شکل 1 می‌بینیم. بازیکن برای اینکه بتواند به مرحله بعد برسد لازم است تمام این دشمن‌ها را از بین برده باشد. در صورتی که یکی از این دشمن‌ها با بازیکن برخورد کند بازیکن را از بین می‌برد.

پس از از بین رفتن یک دیوار ممکن است درب مرحله بعد یا یک قوی‌کننده پشت آن قرار گرفته باشد. درب مرحله بعد تنها پشت یکی از دیوارهای مرحله قرار دارد (درب قهوه ای رنگ در شکل 1). در صورتی که بازیکن تمام دشمن‌های خود در مرحله را از بین برده باشد و بر روی درب مرحله بعد قرار گیرد این مرحله را به پایان می‌رساند.

شکل 2: یک نمونه از قوی‌کننده

همین طور که پیش از این اشاره شد پس از از بین رفتن یک دیوار ممکن است پشت آن یک قوی‌کننده یا ضعیف‌کننده وجود داشته باشد. قوی‌کننده‌ها و ضعیف‌کننده‌ها (مانند قوی‌کننده کنترل زمان انفجار بمب در شکل 1 یا قوی‌کننده قدرت بمب در شکل 2) تغییری در عملکرد بازیکن ایجاد می‌کنند و برخی ویژگی‌ها که باعث قوی‌تر شدن یا ضعیف‌تر شدن بازیکن می‌شود به او اضافه می‌کنند. به عنوان مثال:

- قوی‌کننده افزایش قدرت بمب باعث می‌شود قدرت انفجار بازیکن یک واحد بیشتر شود
- قوی‌کننده کنترل زمان انفجار بمب باعث می‌شود بازیکن بتواند زمان انفجار بمب را در اختیار بگیرد و هر زمان مایل بود بمب را منفجر کند.
- قوی‌کننده افزایش تعداد بمب این امکان را به بازیکن می‌دهد که تعداد بمب‌هایی که می‌تواند در یک لحظه در صفحه داشته باشد یک واحد بیشتر شود.
- قوی‌کننده افزایش سرعت باعث می‌شود که بازیکن در هر واحد زمان یک حرکت بیشتر از قبل بتواند جا به جا شود.
- قوی‌کننده روح باعث می‌شود بازیکن بتواند از سنگ‌ها نیز عبور کند.

به طور مشابه می‌توان ضعیف‌کننده نیز برای بازیکن تعریف کرد. در صورتی که یک بازیکن از روی یک قوی‌کننده یا ضعیف‌کننده عبور کند آن را بدست خواهد آورد.

این بازی می‌تواند به صورت چند نفره نیز انجام شود. در نمونه چند نفره چند بازیکن به صورت رقابتی در نقشه حضور دارند و همزمان به دنبال درب خروج مرحله بعد و از بین بردن دشمن‌ها هستند. در این حالت بمب‌ها بازیکن‌ها را می‌توانند از بین ببرند و به عبارتی بازیکن‌ها می‌توانند با گذاشتن بمب باعث از بین رفتن خود یا سایر بازیکنان هم بشوند. بازیکنی که در این نمونه زودتر از طریق درب مرحله بعد مرحله را به پایان برساند، برنده خواهد شد. به علاوه در این نمونه از بازی قوی‌کننده و ضعیف‌کننده‌هایی که هر بازیکن بدست می‌آورد، مخصوص آن بازیکن خواهد بود.

بخش‌های پروژه درس

در این درس به عنوان پروژه، پیاده‌سازی بازی Bomberman با برخی امکانات بیشتر در نظر گرفته شده است. این پروژه در پنج مرحله پیاده‌سازی خواهد شد که خلاصه این مراحل به صورت زیر است:

1. طراحی، پیاده‌سازی و نمایش نقشه، حرکت بازیکن در نقشه، بمب گذاری در نقشه (نیاز به پیاده سازی انفجار بمب نیست) و ذخیره و بازیابی بازی
2. پیاده‌سازی نمونه یک نفره بازی با تمام امکانات
3. پیاده‌سازی بازی به صورت چند نفره و تحت شبکه
4. امکان اضافه کردن قوی‌کننده و ضعیف‌کننده به بازی در زمان اجرا
5. ذخیره و وضعیت بازی در پایگاه داده و پیاده سازی جدول امتیاز آنلاین تحت وب با استفاده از Django

در هر کدام از مراحل پروژه بسیار توصیه می‌کنیم از ایده‌های خود برای اضافه کردن امکانات بیشتر به پروژه استفاده کنید و از امتیازات بیشتر در ارزشیابی پروژه بهره ببرید. اما در رابطه با اینکه این امکانات مناسب هستند یا خیر و آیا به عملکرد کلی پروژه آسیبی وارد نمی‌کند یا سرپرست‌های تیم خود در ارتباط باشید.

در ادامه به توضیح بخش‌های مختلف پروژه می‌پردازیم.

بخش اول:

در این بخش نقشه بازی را پیاده سازی و نمایش خواهیم داد، به گونه ای که بتوان پس از این مرحله بازیکن در نقشه ظاهر شود و توسط صفحه کلید جا به جا شود و بتواند در نقشه بمب گذاری کند.

در این مرحله پیاده سازی به این صورت خواهد بود که برای شروع بازی بتوانیم بازی را اجرا کنیم و در صفحه اول مشخص کنیم که اندازه تعداد بلوک های صفحه بازی چند در چند باشد. پس انتخاب یک نقشه با ابعاد مشخص شده نمایش داده شود و بازیکن در ابتدا در گوشه بالا سمت چپ نقشه قرار گرفته باشد. در این نقشه سنگ ها با الگو گفته شده قرار خواهند داشت و سایر بلوک ها به صورت تصادفی دارای دیوار یا خالی خواهند بود. سعی کنید نقشه به گونه ای طراحی شده باشد که بازیکن بتواند همواره با بمب گذاری تمام دیوارها را در نهایت از بین ببرد. به عنوان مثال دور تا دور آن دیوار نباشد که امکان از بین بردن آنها بدون از بین رفتن خود بازیکن وجود نداشته باشد (به عنوان مثال هیچ دیواری در ناحیه 2 در 2 در بالا و سمت چپ نقشه نباشد).

در این مرحله پیاده‌سازی حرکت و بمب گذاری بازیکن با استفاده از صفحه کلید نیز انجام می‌شود. کاربر می‌تواند با استفاده از کلید های جهت بازیکن را جا به جا کند و با استفاده از دکمه B یک بمب در محلی که قرار دارد، بگذارد.

در این مرحله نیاز به پیاده‌سازی امکان ذخیره و بازیابی بازی نیز مورد نیاز است. به این منظور در زمان اجرا بازی از فشردن کلیدهای `ctrl+s` برای ذخیره کردن و فشردن کلیدهای `ctrl+o` برای بازیابی بازی استفاده می‌کنیم. در صورتی که کاربر قصد ذخیره کردن بازی را داشت پس از زدن دکمه های مربوط به آن یک پنجره باز خواهد شد که در آن آدرس و نام فایلی که قصد داریم وضعیت بازی را در آن ذخیره کنیم پرسیده می‌شود. پس از ذخیره شدن وضعیت بازی بازیکن به بازی خود ادامه می‌دهد. در صورتی که کاربر در زمان اجرا بازی کلیدهای مربوط به بازیابی را فشار دهد پنجره ای برای انتخاب فایل ذخیره شده از وضعیت بازی نمایش داده می‌شود و پس از انتخاب فایل توسط کاربر، اجرا بازی فعلی قطع شده و بازی جدید از همان وضعیت ذخیره شده در این فایل ادامه پیدا می‌کند.

به علاوه لازم است در صفحه شروع بازی امکان بازیابی بازی‌های ذخیره شده را نیز در نظر گرفته شود.

بخش دوم

همینطور که پیش از این در رابطه با چارچوب این بازی صحبت کردیم در مرحله دوم قصد طراحی نسخه تک نفره بازی را داریم. در این نسخه هدف تکمیل نسخه اصلی بازی با طراحی قوی کننده و ضعیف کننده ها، دشمن‌ها، مرحله های مختلف، بمب گذاری و انفجار بمب و تاثیر بمب بر روی محیط است.

بمب گذاری

اولین امکان مورد نیاز برای اضافه شدن به بازی، امکان بمب گذاری در صفحه، عملکرد و انفجار بمب ها و قوی کننده و ضعیف کننده های مربوط به بمب ها هستند. یک بازیکن در جریان بازی می‌تواند به میزان پارامتر `BombLimit` بمب منفجر نشده در صفحه داشته باشد. این تعداد در شروع بازی برابر 1 است و با دریافت قوی کننده `IncreaseBombs` یا ضعیف کننده `DecreaseBombs` به ترتیب یک واحد افزایش و کاهش پیدا می‌کند، اما کمتر از 1 نمی‌شود. هر بمب به صورت پیش فرض پس از 5 ثانیه منفجر می‌شود مگر اینکه بازیکن قوی کننده `ControlBombs` را در اختیار داشته باشد. در صورتی که بازیکن قوی کننده `ControlBombs` را در اختیار داشته باشد می‌تواند هر زمان که تمایل داشت با فشردن دکمه بمبی در صفحه که زودتر آن را گذاشته است را منفجر کند. در صورتی که یک بمب منفجر شود هر موجودی به جز سنگ که به فاصله حداکثر `BombRadius` در جهت عمودی یا افقی از بمب قرار داشته باشد و دیوار یا سنگی میان این موجود و بمب نباشد، را نابود می‌کند. در صورتی که بمب دیگری در نتیجه انفجار نابود شود، این بمب نیز در لحظه ی نابودی، منفجر می‌شود. امکان اینکه لحظه انفجار بمب با لحظه نابودی موجودات در نتیجه انفجار یکسان نباشد و این لحظه به نسبت فاصله موجود تا بمب متغیر باشد و انیمیشن انفجار بمب نیز تا حد مناسبی این مساله را نشان دهد امتیاز مثبت برای پروژه محسوب می‌شود. مقدار `BombRadius` در آغاز بازی برابر 1 است و با دریافت قوی کننده `IncreaseRadius` و ضعیف کننده `DecreaseRadius` به ترتیب یک واحد بیشتر و کمتر می‌شود، اما کمتر از 1 نمی‌شود.

دشمن ها

امکان دیگر مورد نیاز در این بخش، اضافه کردن دشمن ها است. دشمن ها موجوداتی هستند که پیش از گذر از درب مرحله باید نابود شوند. این موجودات در صورت برخورد با بازیکن او را نابود می کنند. این دشمن ها انواع مختلفی دارند. هر کدام از انواع دشمن ها لازم است که نمایش تصویری منحصر به فردی داشته باشند که بازیکن از روی این نمایش بتواند متوجه نوع دشمن شود. در هر مرحله تعداد دشمن ها به صورت پیش فرض به تعداد کمینه مقدار طول یا عرض صفحه است. به علاوه لازم است که در ابتدا بازی بتوان این تعداد را مشخص کرد. در هر مرحله این تعداد دشمن به طور تصادفی میان انواع دشمنی که امکان حضور در یک مرحله را دارند توزیع می شود. پیاده سازی عملکرد این چهار نوع دشمن از امکانات این بخش به حساب می آید.

- دشمن سطح 1: این دشمن سرعت معادل با نصف سرعتی دارد که بازیکن در ابتدای بازی دارد و کند ترین دشمن محسوب می شود. از نظر حرکت نیز این دشمن در هر لحظه به طور اتفاقی تصمیم می گیرد که در چه جهتی از بین جهت های ممکن حرکت کند. این نوع دشمن در تمام مرحله ها ممکن است ظاهر شود.
- دشمن سطح 2: این دشمن سرعت معادل سرعت دشمن سطح 1 دارد اما همواره در جهتی حرکت می کند که مجموع فاصله عمودی و افقی خود را نسبت به بازیکن کمتر کند. اما این دشمن فاصله واقعی خود با بازیکن را متوجه نمی شود و در مسیرایی دیوار ها و سنگ ها را در نظر نمی گیرد. در صورتی که دو تصمیم مختلف و ممکن برای حرکت این دشمن وجود داشته باشد او یکی از این دو تصمیم را به صورت اتفاقی انجام می دهد. به علاوه در صورتی که با الگو قبل امکان گرفتن تصمیمی نداشت تا ده حرکت به صورت اتفاقی تصمیم به حرکت می گیرد. این دشمن از مرحله دوم می تواند ظاهر شود.
- دشمن سطح 3: این دشمن سرعت معادل با دو برابر دشمن سطح 1 دارد و از نظر تصمیم گیری و حرکت مانند دشمن سطح 2 است. این دشمن از سومین مرحله می تواند ظاهر شود.
- دشمن سطح 4: این دشمن عملکرد مشابه دشمن سطح 3 دارد با این تفاوت که می تواند از دیوار ها و سنگ ها نیز عبور کند. اما نمی تواند از بمب ها عبور کند. این دشمن از چهارمین مرحله می تواند دیده شود.

از مرحله چهارم به بعد در هر مرحله تعداد دشمن ها پنج درصد بیشتر می شود.

قوی کننده ها و ضعیف کننده ها

امکان دیگر مورد نیاز در این بخش از پروژه امتیاز دریافت شده توسط بازیکن است. بازیکن با از بین بردن هر دیوار 10 و با از بین بردن هر دشمن به میزان بیست برابر سطح آن دشمن امتیاز می گیرد. به علاوه در صورتی که بازیکن نتواند در مدت پنج دقیقه مرحله را به پایان برساند در هر ثانیه یک امتیاز از او کم می شود و در صورتی که امتیاز بازیکن کمتر از 0 شود بازیکن از بین می رود.

دیگر امکان این بخش از پروژه پیاده سازی قوی کننده و ضعیف کننده ها است که پیش از این به اکثر آنها اشاره کردیم و در این بخش آنها را مرور می کنیم و یک قوی کننده و ضعیف کننده جدید نیز معرفی می کنیم.

- قوی کننده ها:

- قوی کننده ی BombLimit
- قوی کننده ی IncreaseBombs
- قوی کننده ی IncreaseRadius
- قوی کننده ی ControlBombs
- قوی کننده ی IncreaseSpeed: این قوی کننده پیش از این به طور ضمنی معرفی شد و هر بار دریافت این قوی کننده، میزان سرعت بازیکن که با PlayerSpeed می شناسیم را یک واحد افزایش می دهد و به عبارتی بازیکن در هر واحد زمانی می تواند یک واحد بیشتر حرکت کند.
- قوی کننده ی IncreasePoints: با دریافت این قوی کننده امتیاز بازیکن 100 امتیاز افزایش پیدا می کند.

● ضعیف کننده ها:

- ضعیف کننده ی BombLimit
- ضعیف کننده ی DecreaseBombs
- ضعیف کننده ی DecreaseRadius
- ضعیف کننده ی DecreaseSpeed: هر بار دریافت این ضعیف کننده، PlayerSpeed را یک واحد کاهش می دهد. اما PlayerSpeed کمتر از 1 نمی شود.
- ضعیف کننده ی DecreasePoints: با دریافت این ضعیف کننده امتیاز بازیکن 100 امتیاز کاهش پیدا می کند.

مجموع تعداد قوی کننده ها و ضعیف کننده ها در هر مرحله از بازی دو برابر تعداد دشمن ها در آن مرحله از بازی است، اما نباید از یک سوم تعداد دیوارها در مرحله بیشتر باشد. این تعداد در هر مرحله به طور تصادفی با احتمال مساوی میان قوی کننده ها و ضعیف کننده ها تقسیم می شود.

آخرین مساله در رابطه با قوی کننده ها و ضعیف کننده ها این است که بازیکن تا زمانی که از بین نرفته باشد ویژگی هایی که توسط قوی کننده و ضعیف کننده ها بدست آورده باشد حفظ می کند و این ویژگی ها با رسیدن به مرحله های بعدی از بین نمی رود.

منو بازی

به عنوان دیگر امکان در این بخش از پروژه، در صفحه آغاز بازی نیاز به یک منو داریم که در این منو امکان شروع بازی جدید، تغییر تنظیمات بازی و بازیابی بازی های ذخیره شده در گذشته وجود داشته باشد. به این ترتیب نیاز به ذخیره سازی تمام شرایط بازی در لحظه ذخیره شدن را داریم و در زمان بازیابی بازی دقیقاً از شرایط ذخیره سازی به بعد ادامه پیدا کند.