

キングダムラッシュ説明書

氏名	コウ ショウリン
----	----------

1、ゲームの紹介と使用方法

マウス操作で防御塔を建設およびアップグレードし、入ってくるモンスターを殺し、モンスターが王国に侵入するのを防ぎます。

2、デザインのアイデア

このゲームは、古典的なタワーディフェンスゲームのキングダムラッシュを指し、Easyx プラグインと C 言語を使用して、ゲームを視覚化する目的を達成します。マウスの操作を介して防御塔を構築およびアップグレードして防御線を構築し、タワーディフェンスゲームのコアを達成するよう努めます。可能な限りゲームを復元したいのですが、原作の内容は実に豊富で、ここでは古典的な内容のみを制作しています。

3、機能の説明

プレイヤーのマウス操作を受けることにより、ゲームは対応するコンテンツを提示し、バトルステージの選択とゲーム機能を実現します。その内容の一部を以下に説明します。

3.1、初期化

初期化関数は、一般的なインターフェース画像などのグローバルリソースをインポートするための Start () 関数です。

3.2、バトルステージの選択インターフェース

最初に While (1) ループに入り、アイコンをクリックしてその位置にバトルステージ

ポイントを選択し、構造体 `map.lev` の対応する値を割り当てます。その中で、`map.lev` は、2つの最も重要な関数 `map.Start ()` と `map.Draw ()` を決定します。

3.3、インターフェース切り替えアニメーション

この興味深いデザインは、ゲームをより美しくし、インターフェースの遷移をより調整し、邪魔にならないようにして、プレイヤーが反応する時間を与えます。

3.4、他の機能関数

ゲームには、次のバトルに直接入る、再起動する、バトルステージ選択インターフェースに戻るなどの一連の機能があります。実装は、バトルステージを選択するために必要な小さな関数を `Quit ()` などの大きな関数にパッケージ化することによって実現されます。`Restart ()` このバトルを再開するには、次のバトルに直接移動するには、`map.lev++` と `Restart ()` の組み合わせのみが必要であり、専用の関数はありません。これらの大きな関数は多くの場所で使用されています。

3.5、バトルステージインターフェース

表示されるバトルステージインターフェースは複雑に見えますが、自由にクリックできる防御タワーの場合、それらの座標は事前に対応するファイルに保存されており、読み取りには構造配列と `fscanf(fmap, "%f %f", &tower[i].x, &tower[i].y);` このような短い文を使用しています。

3.6、インターフェースを一時停止

一時停止ボタンをクリックして、ゲームの再開、ゲームの続行、選択したバトルステージの入力、音楽と効果音の制御など、一時停止インターフェースで対応する操作を実現できます。

4、実現プロセスの要点と難しさ

4.1、バッチ操作（工夫した点）

バトルには多くの要素があります。各防御塔、各モンスター、さらに各マップには、対応する png または txt ファイルがあります。ファイル名の形式でインポートすると、コードが長すぎて、動的な実装も難しくなります。したがって、バッチで操作できるメソッドを見つけないのですが、少なくとも画像をバッチでインポートしますが、単純な `loadimage(&image,_T("image%d.png"),i);` はエラーを報告し、実装できません。そのため、そこで、インターネットで対応する方法を探しましたが、関連するチュートリアルがほとんどないことがわかり、最終的に bilibili の目立たないビデオで答えを見つけました。つまり、フォーマット変換方式 `char filename[20]`、`sprintf(filename,"level_%d.png",i)` により、画像のバッチインポートやその他のバッチ操作が実現されます。

4.2、マウス操作の受信とフィードバック（苦労した点）

ゲームの始めに、最初に矢の塔を作るつもりですが、簡単ではありません。防御塔を作るには、マウスで2回連続して操作する必要がある、2つの操作は異なる位置にあります。そのため、最初のクリック後に再びループ内でマウス操作を取得することを

計画しましたが、ループの最後の項目のみがマウス操作を取得できることがわかりました。そこで、防御タワーのメニューが展開されているかどうかを判断するために、変数 `menuopen` を定義しました。その後、防衛塔の紹介ページを表示するために、防衛塔の制作ページにマウスを合わせるという元のゲームの機能を実装するかどうか悩んでいます。しかし、簡単ではないので、クリックした後防衛塔の紹介ページを表示する機能を実装予定でした。ただし、これは実際にはメニューの表示と同じ原理であるため、この機能を実現するために、マウスをホバーするかどうか、およびホバーの位置を決定する変数が定義されます。簡単に言えば、操作の受信とフィードバックは一緒に行う必要はありません。そして、これはプロジェクト全体で実装するのが最も難しい機能だと思います。

4.3、画像の表示（工夫した点）

ゲームはプレイヤーに視覚的なインパクトを与えることであり、画面表示に問題がある場合、そのゲームは面白くないになります。Easyx は画像を消去する機能を備えていないため、ゲームの表示は画像の継続的なカバレッジに依存します。したがって、プレイヤーの操作にリアルタイムのフィードバックを提供するだけでなく、画像の表示順序を調整する必要があります。写真がシームレスに接続されていることを確認してください。そうすれば、関数の配置は非常に重要だと思います。

4.4、攻撃と変位（工夫した点）

モンスターの移動と防衛塔の攻撃には大きな類似点があると思います。モンスターの移動にはターゲットがあります防衛塔の攻撃にもターゲットがあります。そのため、

座標が常にターゲットに近づいている限り、特定のポイントの座標を測定するだけで
モンスターの自動移動を実現し、モンスターの座標を追跡して防御塔追跡攻撃を実現
できます。

5、その他

開発期間：3 週間程度

画像と音楽素材の探す：1 週間程度

ゲームの制作と説明書の作成：2 週間程度

使用ツール：Visual Studio 2019

(**PS:**

- 1、 ゲームで使用される写真や音楽素材は、中国の共有 Web サイトから
ダウンロードして使用されます。
- 2、 EasyXPng.h は私が作成したものではなく、インターネットから参考
して使用しています。
- 3、 ゲームの説明書やコードコメントには多くの専門用語が使われてい
るので、文法上の誤りや説明が不十分な場合はご容赦ください。)