

# ESライブラリ&& ゲームプログラミング

## 2D編 - 第5回 キーボードでキャラクター移動

### キーボード

- ・キーボードの状態は「Keyboard->GetState関数」で取得
- ・「Keyboard->GetState関数」は、KeyboardState型の変数にキーボードの状態を格納する
- ・キーが「押されているか」はIsKeyDown関数
- ・キーが「離されているか」はIsKeyUp関数

#### 概要

キーボードの状態は、Keyboard->GetState関数で取得できます。この関数を実行すると、キーの状態がKeyboardState型の変数に格納されます。

KeyboardState型の変数には、キーの状態を調べるための関数があります。IsKeyDown関数は、引数で指定したキーが押されているかを調べることができます。また、IsKeyUp関数は、引数で指定したキーが離されているかを調べることができます。

特定のキーの状態を調べるには、「Keys\_」ではじまるキー定数またはDirectInputで定義される"DIK\_「ではじまるキーボードデバイス定数を使用します。たとえば'Enter'キーは"Keys\_Enter"、'A'キーは"Keys\_A"です。このように、定数名は"Keys\_"にキーの名称をそのまま加えた名前となっています。

```
// キーボード状態の取得
KeyboardState keyState = Keyboard->GetState();

// キャラクター移動
if(keyState.IsKeyDown(Keys_Left))
    // ' 'が押されているときの処理
if(keyState.IsKeyDown(Keys_Right))
    // ' 'が押されているときの処理
```

### 課題

キーボードでキャラクターが移動するようにしましょう。

(1)キャラクターの座標を格納する変数を宣言します。以下のプログラムをヘッダーファイル"GameMain.h"の適切な場所に追加しましょう。

```
// キャラクター座標
Vector3 ChrPos;
```

キャラクターを表示するスクリーン座標はx座標とy座標の2次元ですが、z値による重なり制御を行えるので、3次元のVector3型で宣言します。

(2)変数を初期化します。以下のプログラムをソースファイル"GameMain.cpp"の"Initialize関数"に追加しましょう。

```
ChrPos = Vector3(0.0f, 0.0f, 0.0f);
```

このプログラムにより、3次元の座標を管理するChrPos変数のx座標、y座標、z座標に0.0が設定されます。画面上の位置では、左上で一番手前に描画されるようになります。

ChrPos変数には、さらにx, y, zという名前の変数があります(メンバ変数といいます)。変数名+メンバ変数名でアクセスします。たとえば、x座標は"ChrPos.x"、y座標は"ChrPos.y"となります。

(3)キャラクターを移動させるプログラムを作成します。以下のプログラムをソースファイル"Game Main.cpp"の"Update関数"に追加しましょう。

```
// キーボード状態の取得
KeyboardState keyboard = Keyboard->GetState();

// キャラクター移動
if(keyboard.IsKeyDown(Keys_Left)) {
    ChrPos.x -= 2.0f; // 左キーが押されたら、ChrPos.xから2を引く
}
```

このプログラムにより、キーボードの' 'キーが押されていると1フレームごとにChrPos.x変数から2が引かれていきます。ChrPos.x変数は、0.0からスタートしているので、0.0 -2.0 -4.0 -6.0...というように変化していきます。このChrPos変数を、キャラクター描画時に指定すると、値が変わるたびにキャラクターの描画位置が変更され、移動しているように見えるという仕組みです。

(4)キャラクターを表示するプログラムを以下のように変更しましょう。

```
// 2D描画
SpriteBatch.Begin();

SpriteBatch.Draw(BGSprite, Vector3(0.0f, 0.0f, 1.0f)); // 背景
SpriteBatch.Draw(CharaSprite, ChrPos); // キャラクター

SpriteBatch.End();
```

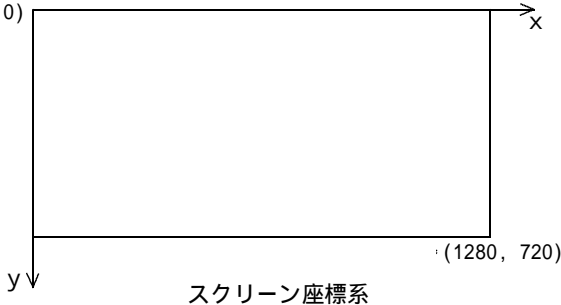
固定値  
変数

背景は座標が固定値になっているため、表示座標が変更されることはありませんが、キャラクターは変数を指定しているので、変数の内容が変更されるたび、表示位置もあわせて変更されます。

(5)プログラムをビルドして実行し、キャラクターを移動できるか確認しましょう。

(6)キャラクターが上下左右に移動できるプログラムを作成しましょう。

ヒント1 : (0,0)



ヒント2 : Keys\_Right, Keys\_Up, Keys\_Down

ヒント3 : キャラクターのx座標はChrPos.x、y座標はChrPos.y