

ESライブラリ&& ゲームプログラミング

荷物勇者編 - 第3回 プレイヤーの移動

キーが押されたか離されたかを判定する

GetKeyState関数で取得できるキーの状態は、「押されている」「押されていない」の2つだけです。押し下げられた瞬間や押し上げられた瞬間という情報を取得することはできません。これらの情報が欲しい場合は、前フレームのキーの状態と現フレームのキーの状態を使って判定します。

前フレームのキーの状態を取得するには、内部処理(Update関数)で現在のキー状態を変数にコピーしておきます。次のフレームになると、それが前フレームのキー状態になります。現在のフレームのキー状態と、コピーしておいたキー状態を比較し、状態が変わっていればキーが「押された」もしくは「離された」が起こったといえます。

キーの状態が前フレームで「押されていない」、現フレームで「押されている」場合、押し下げられた瞬間を表しています。逆に、前フレームでは「押されている」、現フレームでは「押されていない」場合、押し上げられた(放された)瞬間を表しています。

この方法以外に、DirectInputでは入力デバイスに「デバイスバッファ」という領域を設定することができるようになっています。デバイスバッファには、キーの状態に変更があると、その詳細な情報が記録されるようになっています。「状態に変更」というのは、キーが押された瞬間や離された瞬間です。マウスなら、動かされれば「状態に変更」があったことになるので、その詳細な情報が記録されることになります。

ESライブラリでは、デバイスバッファが簡単に読み取れるようになっています。「Keyboard->GetBuffer関数」を実行するだけで、キーボードのデバイスバッファを読み取ることができます。デバイスバッファを読み取ることにより、現在までにキーが押された瞬間や離された瞬間があったかどうかを簡単に判定することができます。

```
// デバイスバッファによるプレイヤー移動
DeviceBuffer KeyBuf = Keyboard->GetBuffer(); // キーボードのデバイスバッファを取得
if(KeyBuf.IsPressed(Keys_Left)) // ' 'キーが押された瞬間だったら、
    PlyPos.x -= 32.0f; // プレイヤー座標を-32.0する
```

課 題

キーボードのカーソルキーが押された瞬間を検出し、プレイヤーが1キャラ分移動するようにしましょう。

(1)プレイヤ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。

```
// プレイヤー座標
Vector3 plyPos;
```

座標は本来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。

```
plyPos = 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plyPos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。

plyPos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"plyPos.x"、y座標は"plyPos.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。

```
// キーボードデバイスバッファの取得
DeviceBuffer keyBuf = Keyboard->GetBuffer();

// プレイヤー移動
if(keyBuf.IsPressed(Keys_Left))
    plyPos.x -= 32.0f;
```

このプログラムにより、キーボードの' 'キーが押された瞬間のみ、plyPos.x変数から32が引かれていきます。plyPos.x変数は、0.0からスタートしているので、0.0 -32.0 -64.0 -96.0...というように変化していきます。このplyPos変数を、キャラクター描画時に指定すると、値が変わるたびにプレイヤ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。

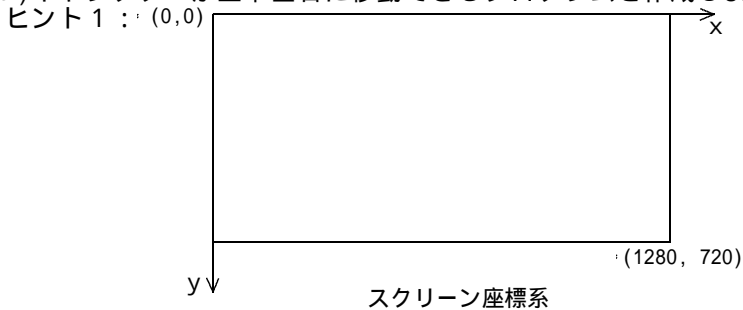
(4)キャラクターを表示するプログラムを以下のように変更しましょう。

```
// スプライト描画
SpriteBatch.Begin();
SpriteBatch.Draw(*plySpr, plyPos); // プレイヤー
SpriteBatch.End();
```

プレイヤーの表示座標に変数を指定したことにより、変数の内容が変更されるたびに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 2 : Keys_Right, Keys_Up, Keys_Down

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