Classe utilitária para teclado em jogos

Este artigo apresenta uma classe utilitária simples para armazenar dados sobre o estado das teclas.

Declaração da classe

```
// INCLUDES
#include <e32keys.h>
#include <e32std.h>

// CLASSES

/**
 * Simples wrapper para os dados do teclado.
 */
class TKeyboardWrapper
{
  public:

    // códigos de teclas não encontrados nos arquivos .h
    enum TKeys {
  EKey0 = 48,
  EKey1,
  EKey2,
  EKey3,
  EKey4,
  EKey5,
  EKey6,
  EKey7,
  EKey8,
  EKey9,
  EKeyShift = 127,
  EKeyStar  = 133
  };

  public:

  /**
   * Construtor padrão.
   */
  TKeyboardWrapper ()
  { Clear (); }


  /**
   * Zerar o estado de todas as teclas.
   */
  void Clear ()
  { Mem::FillZ (iKeys, 256); }


  /**
   * Perguntar se uma tecla está pressionada.
   */
  TBool IsKeyDown (TInt aKey) const
  { ASSERT (aKey >= 0 && aKey < 256); return iKeys [aKey]; }


  /**
   * Determinar que uma tecla está pressionada.
   */
  void SetKey (TInt aKey)
  { ASSERT (aKey >= 0 && aKey < 256); iKeys [aKey] = ETrue; }


  /**
   * Determinar que uma tecla não está mais pressionada.
   */
  void ClearKey (TInt aKey)
  { ASSERT (aKey >= 0 && aKey < 256); iKeys [aKey] = EFalse; }


  /**
   * Verificar se alguma tecla está pressionada.
   */
  TBool AnyKeyDown () const
  {
    for (TInt16 i = 0; i < 256; ++i)
     {
       if (iKeys [i])
 return true;
     }

    return false;
  }

  private:
    TBool iKeys [256];
};
```

Uso

Para se usar efetivamente essa classe, é necessário coletar os dados do teclado em funções que recebam os eventos de entrada, por exemplo `HandleKeyEventL` da `appUI` ou `OfferKeyEventL` das classes de `view`. Um exemplo usando o método `HandleKeyEventL` da classe `appUI`:

```
TKeyResponse CMyAppUi::HandleKeyEventL (const TKeyEvent& aKeyEvent,
TEventCode aType)
{
 if (aType == EEventKeyDown)
  iKeyboard.SetKey (aKeyEvent.iScanCode);
```


(C) Copyright Nokia 2013. All rights reserved.

```
 else
 if (aType == EEventKeyUp)
  iKeyboard.ClearKey (aKeyEvent.iScanCode);

 return EKeyWasConsumed;
}
```

No exemplo, `iKeyboard` é um objeto da classe `TKeyboardWrapper`. Para se consultar o estado das teclas, é preciso usar as constantes definidas pelo tipo enumerado `TStdScanScode`, que está no arquivo `e32keys.h`. Um exemplo de uso:

```
void CMyAppUi::ProcessInput ()
{
  if (iKeyboard [EStdKeyLeftArrow] )
    // seta esquerda foi pressionada

  if (iKeyboard [TKeyboardWrapper::EKey0] )
    // tecla "0" foi pressionada

}
```

É importante não usar por descuido as constantes definidas em `TKeyCode` (que estão no mesmo arquivo `e32keys.h`), pois seu uso não resulta na intenção inicial e também podem causar acessos inválidos no *array* da classe (pois alguns são maiores que 256).

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.