SFML i przechwytywanie znaków (sheadovas/poradniki/goto/sfml-i-przechwytywanie-znakow/)

Wrz 18, 2015 / goto (sheadovas/category/poradniki/goto/)

O przechwytywaniu znaków z klawiatury i tworzeniu własnego „TextBox'a" w SFML.

Z pewnością wielu z Was chciałoby umożliwić graczowi podania np. swojego pseudonimu do tablicy wyników, albo do stworzenia profilu gracza o dowolnej nazwie (aby nie nazywać S*lot 1*, *Slot 2*, ...). Jednakże jak się do tego zabrać?

W tym wpisie zajmiemy się przygotowaniem prostego kodu umożliwiającego użytkownikowi na wprowadzeniu tekstu z klawiatury, dodatkowo stworzymy ograniczenie ilości znaków jakie można wprowadzić oraz zabezpieczymy się przez wprowadzaniem niepożądanych przez nas znaków (np. znaków białych).

przechwytywanie-1

(https://i1.wp.com/www.shead.ayz.pl/wp-content/uploads/2015/09/przechwytywanie-1.png)

Implementacja

Proste przechwytywanie znaków

SFML umożliwia nam przechwytywanie znaków z klawiatury przy użyciu zdarzenia: TextEntered i najprostszy najbardziej ogólny kod służący do przechwytywania znaków może wyglądać mniej więcej w ten sposób.

Najprostsze przechwytywania znaków do zmiennej C++

```
...
sf::Event event;
std::string string = ""; // <- zmienna przechowująca wpisany tekst

... // <- pętla pollEvent
if (event.type == Event::TextEntered)
{
        if (event.text.unicode < 128)
                string += static_cast<char>(event.text.unicode);

}
```

To co tutaj widzimy to sprawdzenie czy wciśnięty znak ma wartość mniejszą niż 128 (czyli interesujące nas znaki, można zmniejszyć ten zakres, zalecam zajrzeć do tabeli Unicode aby dowiedzieć się jakie znaki mają jaki kod), następnie rzutujemy go na char (a zatem na ASCII) i dopisujemy go do obiektu string.

Oczywiście fajnie by było jeszcze widzieć, że faktycznie coś się dzieje po wciśnięciu klawiszy, więc po drobnych modyfikacjach kod prezentuje się następująco:

Wpisywanie tekstu i jego wyświetlanie C++

```
sf::Text text;
... // <- ustawienie czcionki, pozycji, etc

std::string string = "";
sf::RenderWindow window(...);

while(window.isOpen())
{
 sf::Event event;

 while(window.pollEvent(event)
 {
 if (event.type == Event::TextEntered)
 {
 if (event.text.unicode < 128)
 string += static_cast<char>(event.text.unicode);
 text.setString(string);
 }
 }
 ... // <- czyszczenie, rysowanie innych ewentualnych obiektów
 window.draw(text); // rysowanie tekstu
 ...
 window.display();
}
```

(pełny kod: gotowy do skopiowania, wklejenia i uruchomienia znajduje się na dole wpisu)

Ograniczenie ilości znaków

Dalsze modyfikacje są dość banalne i jak łatwo się domyślić tak można dość łatwo dodać maksymalną ilość znaków np w ten sposób:

```cpp
std::size_t maxTextLength = 5;

...

if (event.type == Event::TextEntered)
{
 if (event.text.unicode < 128 && string.size() < maxTextLength)
 string += static_cast<char>(event.text.unicode);
 text.setString(data);
 }
}
```

Usuwanie znaków

Użytkownik może oczywiście zechcieć usunąć znak, całość można dość ładnie wszystko obsłużyć także wewnątrz *TextEntered*.

Usuwanie znaków

```cpp
... // <- wnętrze TextEvent

if (event.text.unicode == 'b' && string.size() > 0)
{
 string.erase(string.size() - 1, 1);
 text.setString(string);
}

else if (event.text.unicode < 128 && string.size() < maxTextLength)
{
 string += static_cast<char>(event.text.unicode);
 text.setString(string);
}
```

Wyłącznie standardowe znaki

Aby dopuścić do używania wyłącznie „normalnych” znaków, tzn liter i cyfr należy wprowadzić dozwolone przedziały: [48,57] + [65,90] + [97,122]. Aby dowiedzieć się skąd wzięły się te przedziały polecam zajrzeć na stronę z tabelą kodów unicode (http://unicode-table.com/en/).

Ograniczenie do standardowych znaków

```cpp
... // <- wnętrze TextEntered

sf::Uint32 unicode = event.text.unicode;

if (((unicode >= 48 && unicode <= 57) || (unicode >= 65 && unicode <= 90) ||
 (unicode >= 97 && unicode <= 122)) && string.size() < maxTextLength)
{
 string += static_cast<char>(unicode);
 text.setString(string);
}
```

Kompletny kod

Dla leniwych poniżej przedstawiam kod zawierający wszystkie pokazane w tym wpisie featery. Kod jest gotowy do skopiowania, wklejenia i uruchomienia.

Kompletny kod (copy & paste & run)

```cpp
```

```cpp
#include <SFML/Graphics.hpp>
#include <string>


int main()
{
 std::size_t maxTextLength = 5;

 std::string string;
 sf::Text text;

 sf::Font font;
 font.loadFromFile("Ubuntu-R.ttf");

 text.setFont(font);
 text.setCharacterSize(40);
 text.setPosition(10, 10);

 sf::RenderWindow window(VideoMode(600, 80), "Przechwytywanie znakow w SFML", Style::Close);
 while (window.isOpen())
 {
 sf::Event event;

 while (window.pollEvent(event))
 {
 if (event.type == Event::Closed)
 window.close();

 else if (event.type == Event::TextEntered)
 {
 sf::Uint32 unicode = event.text.unicode;

 if (unicode == 'b' && string.size() > 0)
```

Posłowie

Dajcie znać co myślicie o tym wpisie, jeżeli chcielibyście rozwinięcia tego co zrobiliśmy lub chcielibyście zobaczyć więcej rzeczy tego typu (ciekawostek SFML, chociaż może być to także coś innego) to dajcie znać w komentarzach.

Code ON!