```cpp
#include <sstream>
#include <Windows.h>
//custom window process

LRESULT CALLBACK WndProc(HWND hwnd, UINT msg, WPARAM wParam, LPARAM lParam)
{
        switch (msg)
        {
        case WM_CLOSE:PostQuitMessage(69);
        break;
        //cambia il nome alla finestra premendo f
        case WM_KEYDOWN:
                if (wParam == 'F')
                {
                        SetWindowText(hwnd, "Roberto");
                }
        case WM_KEYUP:
                if (wParam == 'F')
                {
                        SetWindowText(hwnd, "Milone");
                }
                break;
                //scrive le coordinate del mouse sulla finestra
        case WM_LBUTTONDOWN:
        {
                const POINTS pt = MAKEPOINTS(lParam);
                std::ostringstream oss;
                oss << "(" << pt.x << "," << pt.y << ")";
                SetWindowText(hwnd, oss.str().c_str());
        }
        break;
        }

        return DefWindowProc(hwnd, msg, wParam, lParam);
}

int CALLBACK WinMain(HINSTANCE hInstance,
        HINSTANCE hPrevInstance,
        LPSTR lpCmdLine,
        int nCmdShow)
{
        const auto pClassName = "hw3dbutts";
        //register window class
        WNDCLASSEX wc{ 0 };
        wc.cbSize = sizeof(wc);
        wc.style = CS_OWNDC;
        wc.lpfnWndProc = WndProc;
        wc.cbClsExtra = 0;
        wc.cbWndExtra = 0;
        wc.hInstance = nullptr;
        wc.hIcon = nullptr;
        wc.hCursor = nullptr;
        wc.hbrBackground = nullptr;
        wc.lpszMenuName = nullptr;
        wc.lpszClassName = pClassName;
        wc.hIconSm = nullptr;
        RegisterClassEx(&wc);
        //create window instance
        HWND hWnd = CreateWindowEx(0, pClassName,
                "Happy Hard Window", WS_CAPTION | WS_MAXIMIZEBOX
                | WS_SYSMENU, 200, 200, 640, 480, nullptr, nullptr, hInstance, nullptr);
        //show the damn window
        ShowWindow(hWnd, SW_SHOW);
        //message pump
```

```cpp
    MSG msg;
    BOOL getResult;
    while ((getResult= GetMessage(&msg, nullptr, 0, 0) > 0))
    {
        TranslateMessage(&msg);
        DispatchMessage(&msg);
    }
    if (getResult == -1)
    {
        return -1;
    }
    else
    {
        return msg.wParam;
    }
    return 0;
}
```