

Installasjon av SDL (1)

- Last ned SDL x86 Windows binaries
 - It's Learning: F04 – Grafikkprog. → ressurser
→ SDL 2.0...zip
 - GitHub: <http://goo.gl/KIFn8>
- Kjører du Linux eller OSX må du bygge selv
 - <http://www.libsdl.org/tmp/SDL-2.0.tar.gz>
 - Evt. <http://www.libsdl.org/hg.php>
- Pakk ut (unzip) til et egnet sted (ikke Desktop)
 - eks. *C:/Program Files (x86)/SDL-2.0.0-6799/*

Installasjon av SDL (2)

- Gå inn i installasjonsmappen
 - eks. *C:/Program Files (x86)/SDL-2.0.0-6799/*
- Gå deretter inn i */lib/Release*
 - Kopier/klipp ut *SDL.dll*
- Gå til *C:/Windows/system*
 - Lim inn *SDL.dll*
- Evt. kan du legge *SDL.dll* i en hvilken som helst mappe. Men da må du legge denne mappen inn i miljøvariabelen *PATH*, likt dere gjorde med *bin*-mappen til Java i 1. klasse.
- Evt. kan du kopiere *dll*-filen inn i root-mappen til hvert eneste nye VS12-prosjekt som bruker SDL

User-Defined Macros i VS12 (1)

- Et navn/verdi-par som kan brukes som en build-time variabel.
- Blir lagret unikt per bruker, per maskin.
- Eksempel, vi lager en ny Macro
 - Macro/variabel-navn: PG4400_SDL2_PATH
 - Verdi: C:/Program Files (x86)/SDL-2.0.0-6799
- Vi kan da sette at vi skal inkludere filene i mappen i vårt prosjekt: `$(PG4400_SDL2_PATH)/include`
 - Som da blir gjort om til «C:/Program Files (x86)/SDL-2.0.0-6799/include» ved build

User-Defined Macros i VS12 (2)

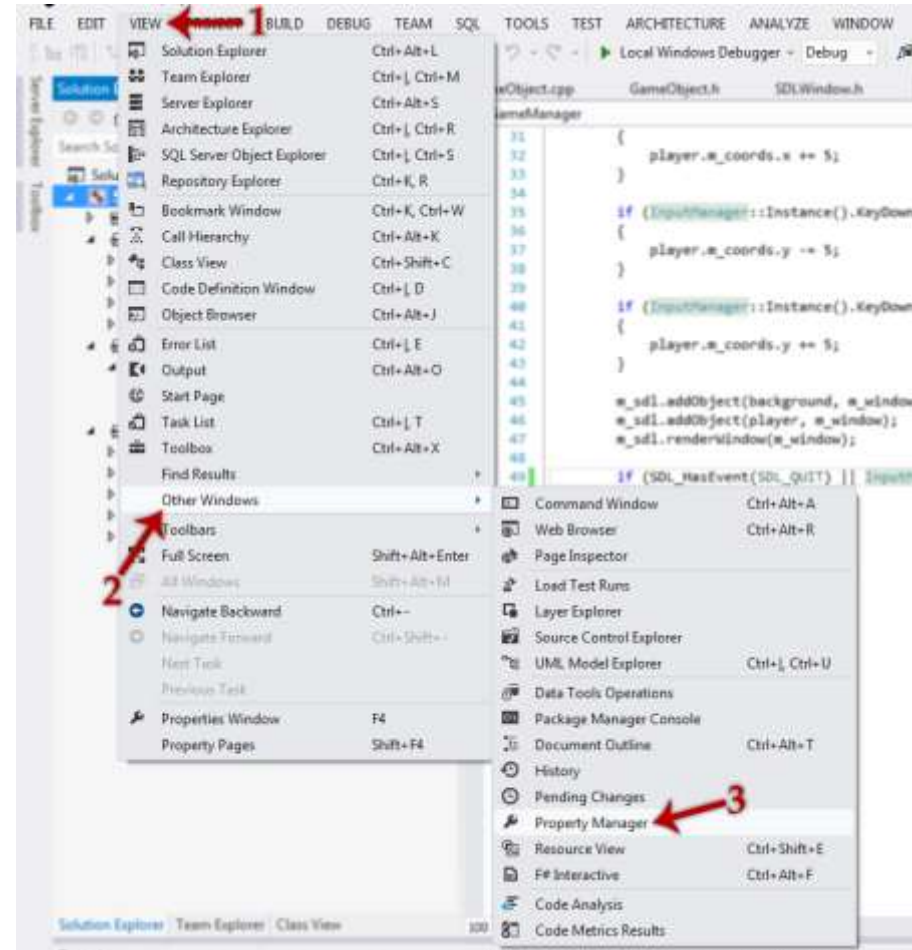
- Hvilken fordel gir dette?
- Gitt følgende scenario:
 - Arne-Christian har SDL installert i «*D:/SDL/*»
 - Trym har installert SDL i «*C:/Users/Trym/Desktop/SDL/*»
 - De har begge «hardkodet» path-en i Visual Studio (ikke Macro)
- Hva skjer når Arne-Christian skal samarbeide med Trym?
 - Det kompilerer ikke! // *Can't find path ...*
 - Trym må endre det manuelt hver gang han får en ny versjon
 - Git/SVN tar ikke høyde for dette
 - Husk, jeg skal rette 22 innleveringer med potensielt 22 forskjellige paths!

SDL User-Defined Macro (1)

- Finn VS12 Property Manager:

- Åpne et hvilket som helst VS12-prosjekt
 - Lag evt. et nytt!

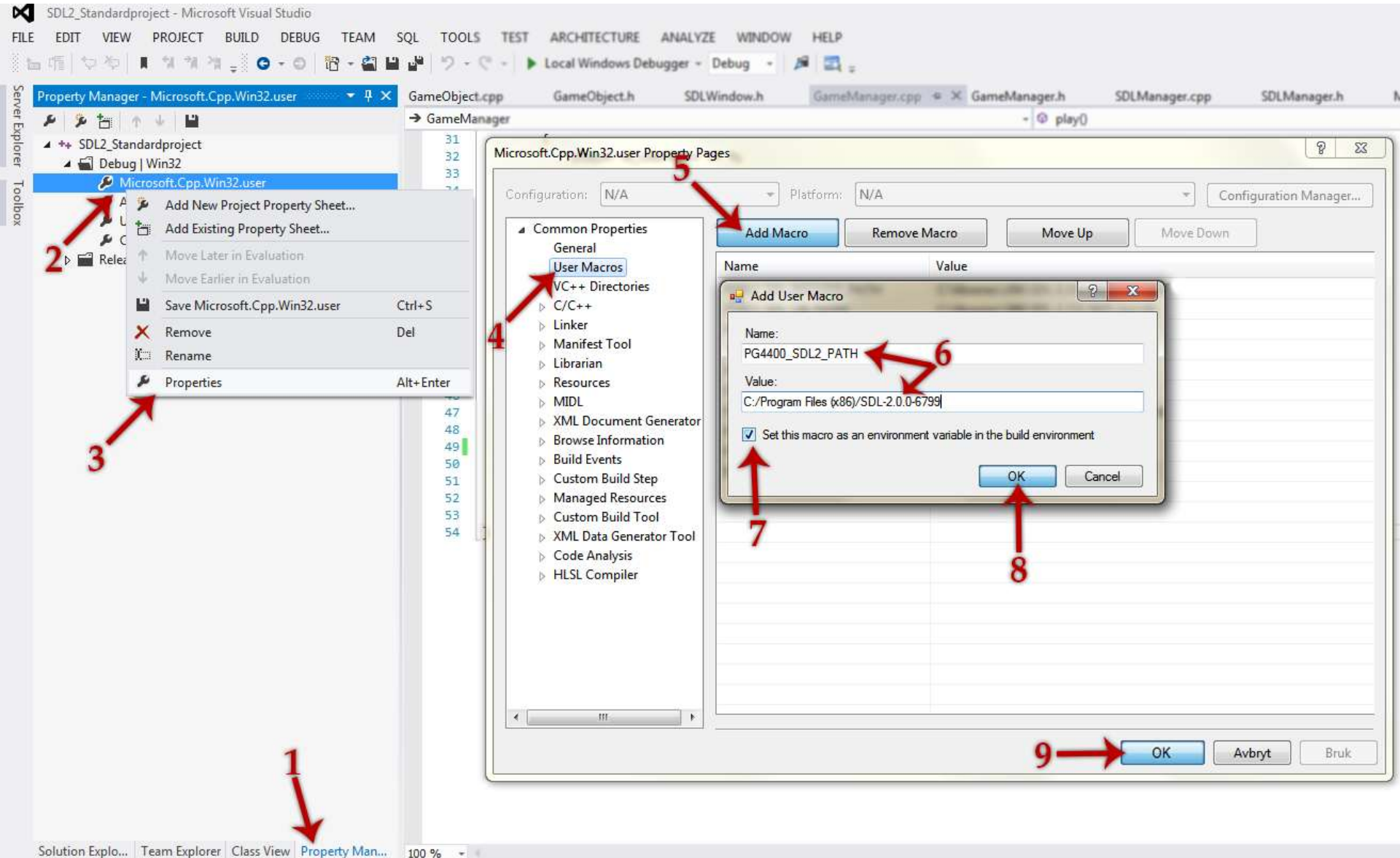
- Trykk View → Other Windows → Property Manager



SDL User-Defined Macro (2)

- Gå i Property Manager
- Finn «Microsoft.Cpp.Win32.user» i «Debug | Win32»
- Høyreklikk → Properties
- Til venstre: Common Properies → User Macros
- I hoved-vinduet: [Add Macro]
 - Name: PG4400_SDL2_PATH
 - Value: <din SDL-path> eks: *C:/Program Files (x86)/SDL-2.0.0-6799*
 - Huk av «Set this Macro as an environment variable ...»

SDL User-Defined Macro (3)



VS12: Opprett et SDL-prosjekt (1)

- Lag et nytt vanlig konsollprosjekt
- Sørg for å være i «Solution Explorer»
 - Ikke «Property Manager»
- Høyreklikk prosjektnavnet (under solution)
- Velg Properties
- Sørg for at «Debug» eller «Active(Debug)» er satt som «Configuration»

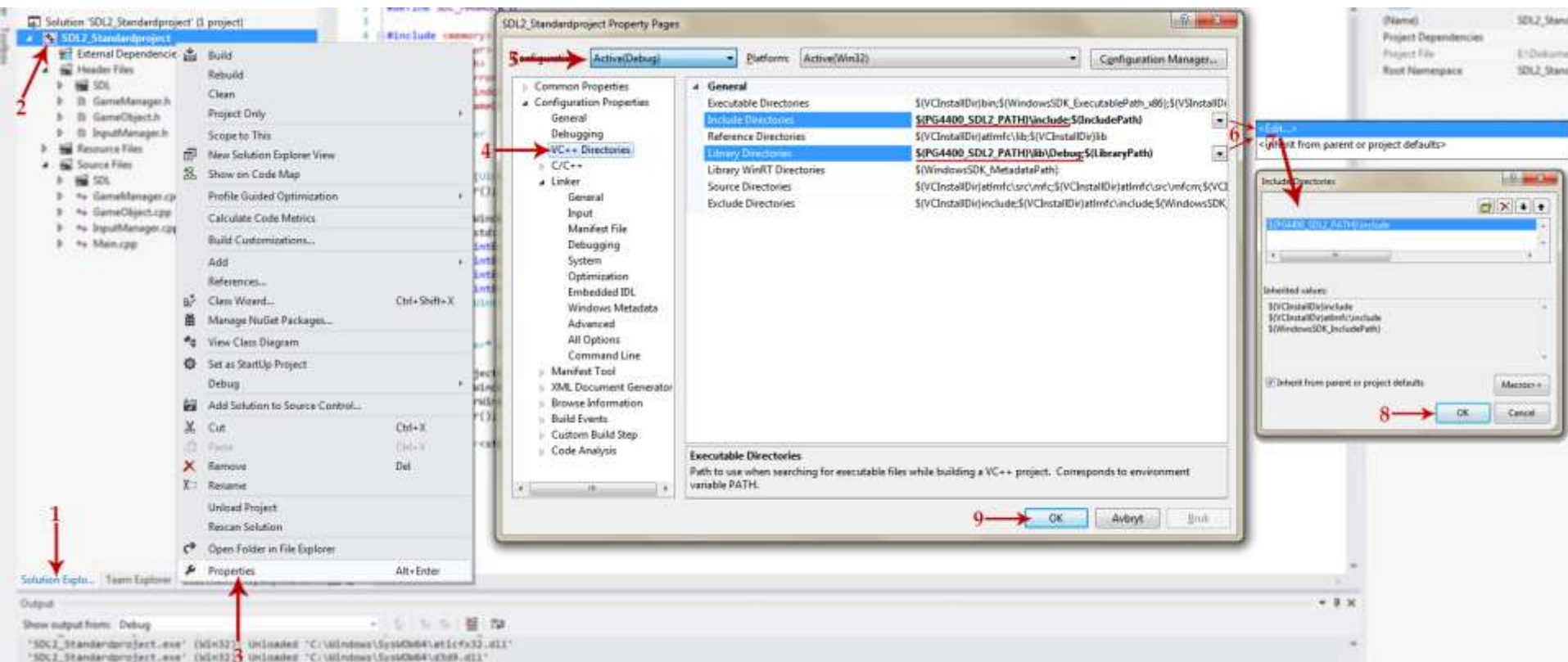
VS12: Opprett et SDL-prosjekt (2)

- Gå til «Configuration Properties» → «VC++ Directories»
- Trykk på «Include Directories», deretter pilen som dukker opp, og velg <Edit...>
- Legg til «\$(PG4400_SDL2_PATH)**include**» og trykk OK
- Gjør det samme med «Library Directories», men legg til «\$(PG4400_SDL2_PATH)**lib\Debug**» isteden.

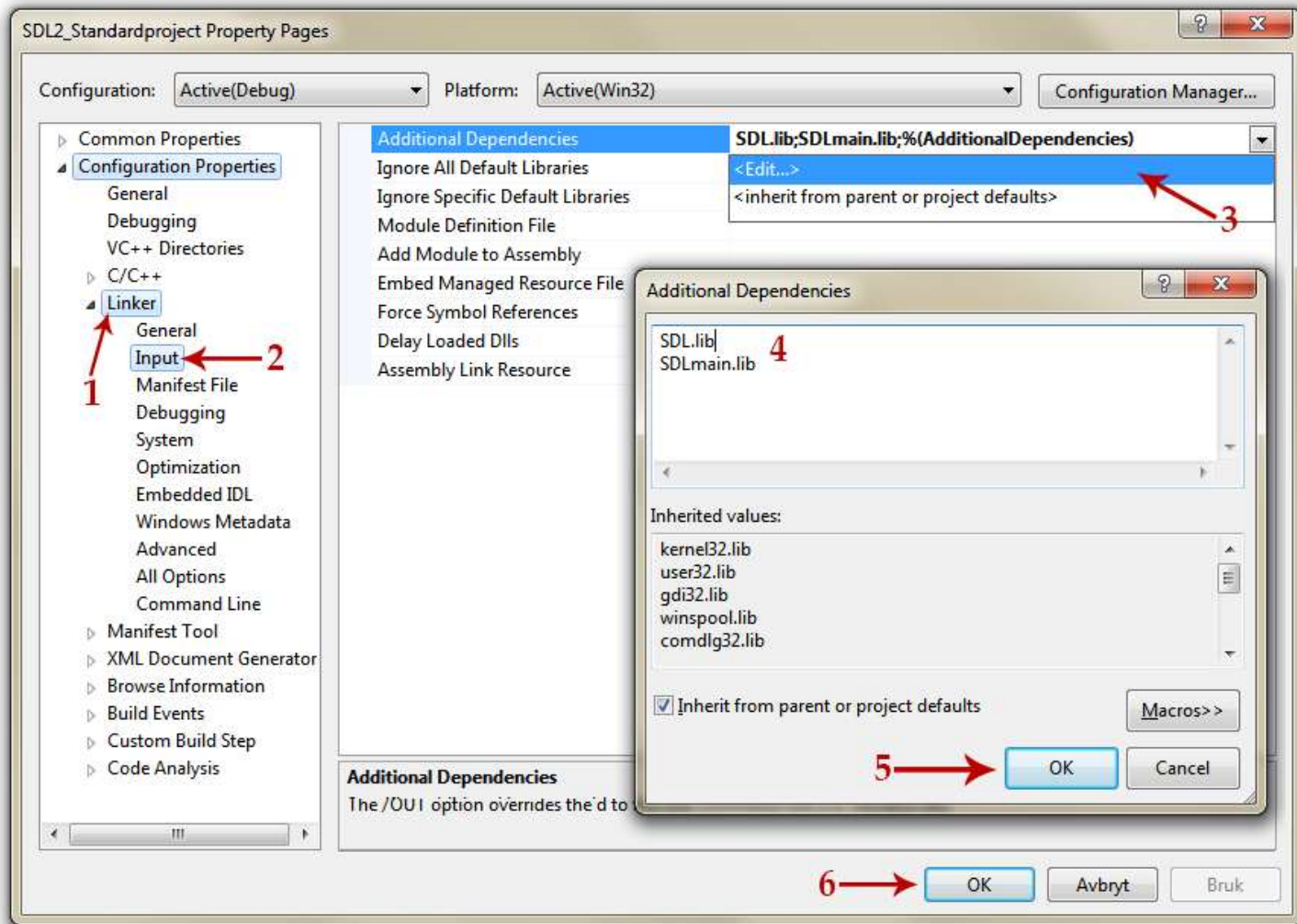
VS12: Opprett et SDL-prosjekt (3)

- Gå så til «Configuration Properties» → «Linker»
→ «Input»
- Trykk på «Additional Dependencies» , deretter pilen som dukker opp, og velg <Edit...>
- Legg till «SDL.lib» OG «SDLmain.lib», trykk OK
- Trykk «Bruk»/«Use»
- Du bør gjøre det samme for Release, men bruk i så fall «\$(PG4400_SDL2_PATH)\lib\Release» isteden

VS12: Opprett et SDL-prosjekt (4)



VS12: Opprett et SDL-prosjekt (5)



Test om det funker (1)

Lag en ny exception

SDL_Error.h

```
#ifndef SDL_ERROR_H
#define SDL_ERROR_H

#include <exception>
#include <string>

class SDL_Error : public std::exception
{
public:
    SDL_Error();
    SDL_Error(const std::string&);
    virtual ~SDL_Error() throw();
    virtual const char* what() const throw();
private:
    std::string m_message;
};

#endif
```

SDL_Error.cpp

```
#include <SDL.h>
#include "SDL_Error.h"

SDL_Error::SDL_Error() : exception()
{
    m_message = SDL_GetError();
}

SDL_Error::SDL_Error(const std::string& msg)
    : exception()
{
    m_message = msg;
}

SDL_Error::~~SDL_Error() throw()
{
}

const char* SDL_Error::what() const throw()
{
    return m_message.c_str();
}
```

Test om det funker (2)

Lag en ny Main

Main.cpp

```
#include <iostream>
#include "SDL_Error.h"

int main(int argc, char* argv[])
{
    try
    {
        if (SDL_Init(SDL_INIT_VIDEO))
        {
            std::stringstream msg;
            msg << "Could not initialize SDL: " << SDL_GetError();

            throw SDL_Error(msg.str());
        }

        return EXIT_SUCCESS;
    }
    catch(const SDL_Error& err)
    {
        std::cerr << "An SDL error occured: "
                  << err.what() << std::endl;
    }

    return EXIT_FAILURE;
}
```

Om det builder, kjører
og returnerer 0:



Bilde fra: <http://memeorama.com/meme/great-success-borat-meme/>