GCC Einleitung

GCC steht für GNU Compiler Collection mit der man u.a. C/C++, Java, Fortran, Assembler und zahlreiche weitere Sprachen in ausführbaren Code übersetzen kann. GCC existiert für die verschiedensten Platformen u.a. Linux, Windows. Mit dieser Collectionen können auch Cross-Compilationen durchgeführt werden.

GCC Beispiel

$ gcc -g -O2 -I/usr/local/include/SDL -L/usr/local/lib -lSDL -Wall -o programm.exe programm.c

Für C++ Programme muss man anstatt gcc, g++ aufrufen!!!

In diesem Beispiel wird ein Programm übersetzt, welches die SDL Library verwendet. Es sind alle Warnungen angeschaltet, Debuginformationen hinzugefügt und alle Warnungen zur Compilezeit eingeschaltet. Siehe unten!

GCC Optionen

Die folgende Tabelle beinhaltet alle wichtigen Optionen für GCC um C/C++ in ausführbare Programme zu übersetzen.

Option	Beschreibung
-c	Erstellt Objektdateien aus den Quellen.
-o	Linkt die Objektdateien zu einem ausführbaren Programm.
-g	Generiert den Code mit zusätzlichen Informationen um das debuggen zu ermöglichen.
-O	Aktiviert Optimierungsmethoden, drei Stufen entscheiden über die Optimierung: -O1, -O2, -O3 der Grad der Optimierung entscheided über die Größe des Programms und dessen Ausführungszeit. Standartwert ist -O2.
-I	Diese Option bindet zur Compilierzeit weitere Verzeichnisse hinzu, die für #include <...> Anweisungen durchsucht werden sollen. Diese Option sollte vor einer manuellen Angabe des Pfades im Include-Statement verwendet werden.
-L	Gibt weitere Verzeichnisse an in denen sich Bibliotheken (*.so, *.a) befinden. Verweis auf -l.
-l	Diese Option linkt beim linken der Objektdateien weitere Bibliotheken zum Programm hinzu.
-Wall	Zeigt alle Warnungen an. Warnungen sind nicht zu verachten und sollten wenn möglich verhindert werden.

GCC/SDL Beispiel

```
#include <stdio.h>
#include <SDL.h>

int main( int argc, char **argv )
{
   SDL_Surface * surface;
   SDL_PixelFormat *fmt;
   Uint8 i;

   SDL_Init( SDL_INIT_VIDEO );
   SDL_WM_SetCaption( "SDL Hello World", NULL );
   surface = SDL_SetVideoMode( 640, 480, 32, SDL_HWSURFACE |
                                    SDL_HWPALETTE |
                                    SDL_DOUBLEBUF );
   SDL_Rect rectangle;
   rectangle.x = 170; rectangle.y = 170;
   rectangle.w = 300; rectangle.h = 140;

   fmt = surface->format;

   SDL_LockSurface(surface);
   SDL_FillRect( surface, &rectangle, SDL_MapRGB( fmt, 255, 0, 0) );
   SDL_UpdateRect( surface, rectangle.x, rectangle.y, rectangle.w, rectangle.h );
   SDL_UnlockSurface(surface);

   SDL_Event event;
   while(SDL_WaitEvent(&event)) {
      if( event.type == SDL_KEYUP )
         break;
   }
   SDL_Quit();
}
```

Listing 1: SDL Beispiel fürs Verständnis.

Dieses einfache Beispiel initialisiert das SDL Framework und zeichnet ein Rechteck in die Mitte des Fensters. Beenden kann man das Programm durch das drücken einer beliebigen Taste auf der Tastatur.

$ gcc `sdl-config –cflags` `sdl-config –libs` -o beispiel_sdl beispiel_sdl.c

Führt 'sdl-config –libs' manuell in einem Terminal aus und ihr erkennt was es macht.