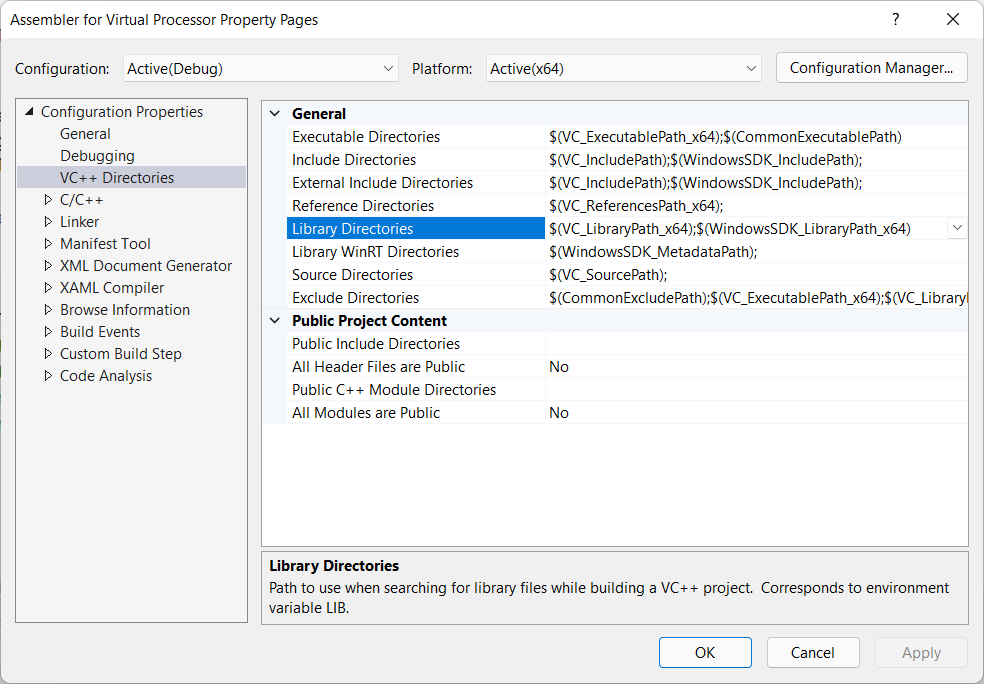
1. Copy the following .dll files under C:\Windows\system32 path.

glu32.dll  
 glut32.dll  
 opengl32.dll

* For step 2 and 3. In Visual Studio go to Solution properties and find the path for libraries there:



1. Copy the following headers under C:\Program Files (x86)\**Windows Kits\10\Include\10.0.18362.0\um** path.

gl.h  
 glaux.h  
 glu.h  
 glui.h  
 glut.h

1. Copy the following libraries files under C:\Program Files (x86)\**Windows Kits\10\Lib\10.0.18362.0\um** path.

Glaux.lib  
 Glu32.lib  
 Glui32.lib  
 Glu32.lib  
 Opengl32.lib

1. Download Visual Studio Community 2019 from the following link:

<https://visualstudio.microsoft.com/vs/community/>

Install Visual Studio Community 2019.

Chose workloads - Desktop Development with C++  
  
 Modifying — Visual Studio Community 2019 — 16.5.1 
Workloads 
web & Cloud (4) 
Individual components 
Language packs 
ASP. NET and web development 
Build web applications using ASP.NET Core, ASP.NETI 
HTML/JavaScript, and Containers including Docker support. 
Phython development 
Editing, debugging, interactive development and source 
control for Python. 
Desktop & Mobile (5) 
.NET desktop development 
Build WPF, Windows Forms, and console applications using 
Visual Basic, and with .NET Core and .NET... 
Universal Windows Platform development 
Create applications for the Universal Windows Platform 
with VB, or optionally 
Location 
Files (x86)\Microsoft Visual Studio\201 g\Community 
s 
Installation locations 
Azure development 
Azure SDKs, tools, and projects for developing cloud apps 
and creating resources using .NET Core and .NET.. 
Node.js development 
Build scalable network applications using Node.js, an 
asynchronous event-driven JavaScript runtime. 
Desktop development with * 
Build modern C+ + apps for Windows using tools of your 
choice, including MSVC, Clang, CMake, or MS8uiId. 
Mobile development with .NET 
Build cross-platform applications for iOS, Android or 
Windows using Xamarin. 
x 
Installation details 
> Visual Studio core editor 
v Desktop development with C++ * 
Included 
C++ core desktop features 
IntelliCode 
Optional 
MSVC vi42 - VS 2019 C++ x64/x86 build tools 
Windows 10 SDK (10.0.18362.0) 
Just-ln-Time debugger 
C++ profiling tools 
C++ CMake tools for Windows 
C++ ATL for latest v142 build tools (x86 & x64) 
Test Adapter for Boost.Test 
Test Adapter for Google Test 
Live Share 
C++ AddressSanitizer (Experimental) 
C++ MFC for latest v142 build tools (x86 & x64) 
C++/CLI support for v142 build tools (14.25) 
C++ Modules for v142 build tools (x64/x86 — ex... 
C++ Clang tools for Windows (9.0.0 - x64/x86) 
Total space required 
647 

5. Create a new Project

Create a new project 
Choose a project template With code scaffolding 
to get started 
Continue without code 

6. Empty project template

Empty Project 
Start from scratch with C++ for Windows. Provides no starting files. 
Windows 
Console 

7. Right click on project -> Properties -> Linker -> Input

Project I 
1. Right CLICK 
Header Files 
Resource Files 
Source Files 
Solution 'Projectl' (I of I project) 
Cut 
3uild 
*ebuild 
Clean 
Analyze and Code Cleanup 
Project Only 
Retarget Projects 
Scope to This 
New Solution Explorer View 
Build Dependencies 
Class Wizard... 
Manage NuGet Packages... 
Set as Startup Project 
Debug 
Source Control 
Re mcn.'e 
Rename 
Unload Project 
Load Project Dependencies 
Rescan Solution 
Display Browsing Database Errors 
Clear Browsing Database Errors 
Open Folder in File Explorer 
Projectl Property Pages 
Configuration: Active(Debug) 
Configuration Properties 
General 
Advanced 
Debugging 
VC++ Directories 
Platform: 
Output File 
Show Progress 
Version 
Enable Incremental Linking 
Linker 
General 
Input 
3. Click Input under Linker 
Ctrl+Shift+X 
Ctrl+X 
Ctrl+V 
Manifest File 
Debugging 
System 
Optimization 
Embedded IDL 
Windows Metadata 
Advanced 
All Options 
Command Line 
Manifest Tool 
XML Document Genera 
Browse Information 
Build Events 
Custom Build Step 
Register Output 
Per-user Redirection 
Additional Library Directories 
Link Library Dependencies 
Use Library Dependency Inputs 
Link Status 
Prevent 011 Binding 
Treat Linker Warning As Errors 
Force File Output 
Create Hot Patchable Image 
Specify Section Attributes 
Output File 
The 'OUT option overrides the defal 
Properties 
2. Click Properties 

8. Click on Edit Additional Dependencies and add the following libraries.

Glu32.lib  
 Opengl32.lib  
 Glaux.lib  
 Legacy\_stdio\_definitions.lib  
  
 Additional Dependencies 
nore All Default Libraries 
Additional Dependencies 
Blu32.Iib 
opengB2.Iib 
glaux.lib 
legacy_stdio definitions.lib 
stdio 

9. Add the glos.h and ".c" files in the project.

Right click -> Add -> Existing Item or New Item to add a file.  
  
 For .h file click on Header Files folder and Source Files for .c files.  
  
   
  
 Project I 
References 
External Dependencies 
Sou 
Ctrl+Shift+X 
Ctrl+X 
Ctrl+C 
Ctrl+V 
New Item... 
Existing Item... 
New Filter 
Class... 
Resource... 
Ctrl+Shift+A 
Shift+AIt+A 
Class Wizard... 
Scope to This 
New Solution Explorer View 
Cut 
Copy 
Delete 
Rename 
Properties 

Solution 'Projectl' (I of I project) 
Project I 
31 
References 
External Dependencies 
Header Files 
GLOS.H 
Resource Files 
Source Files 
C Patrat.c 