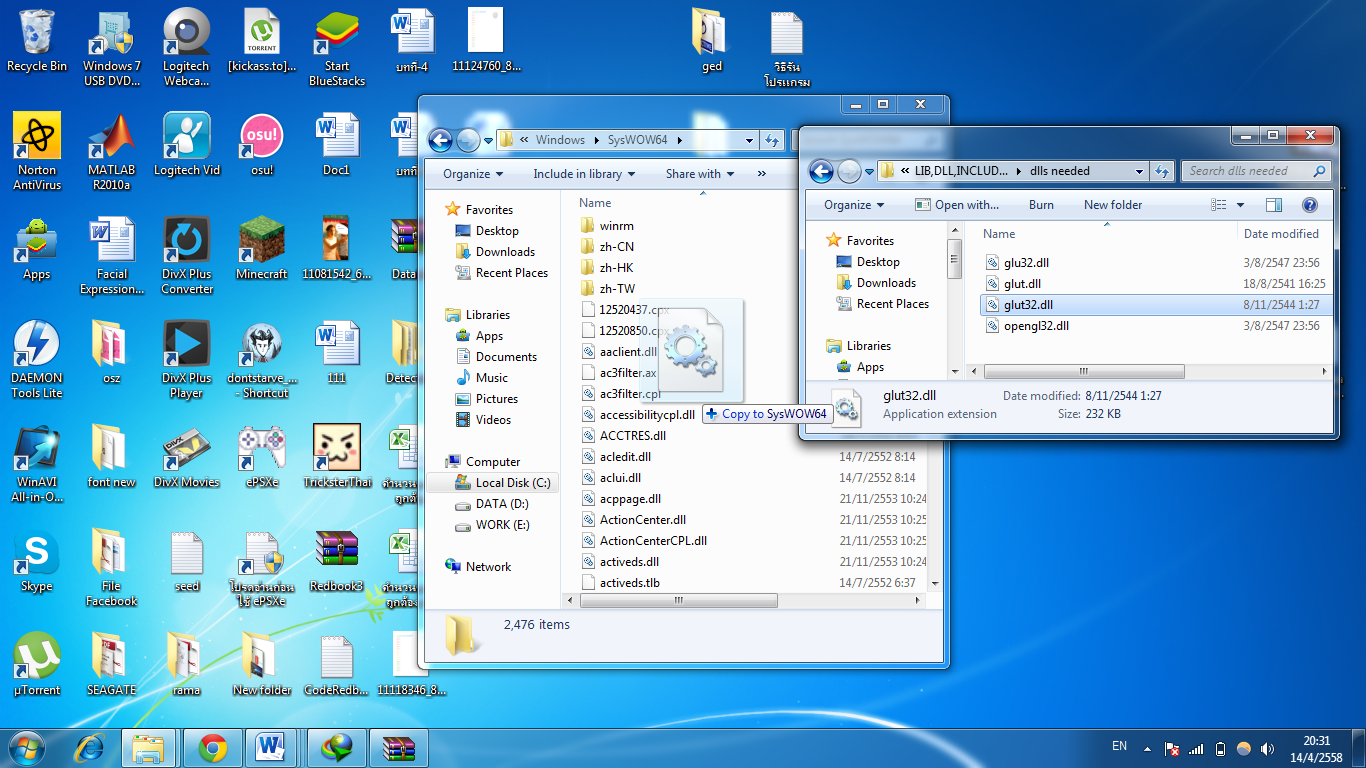
**ภาคผนวก**

**OpenGL**

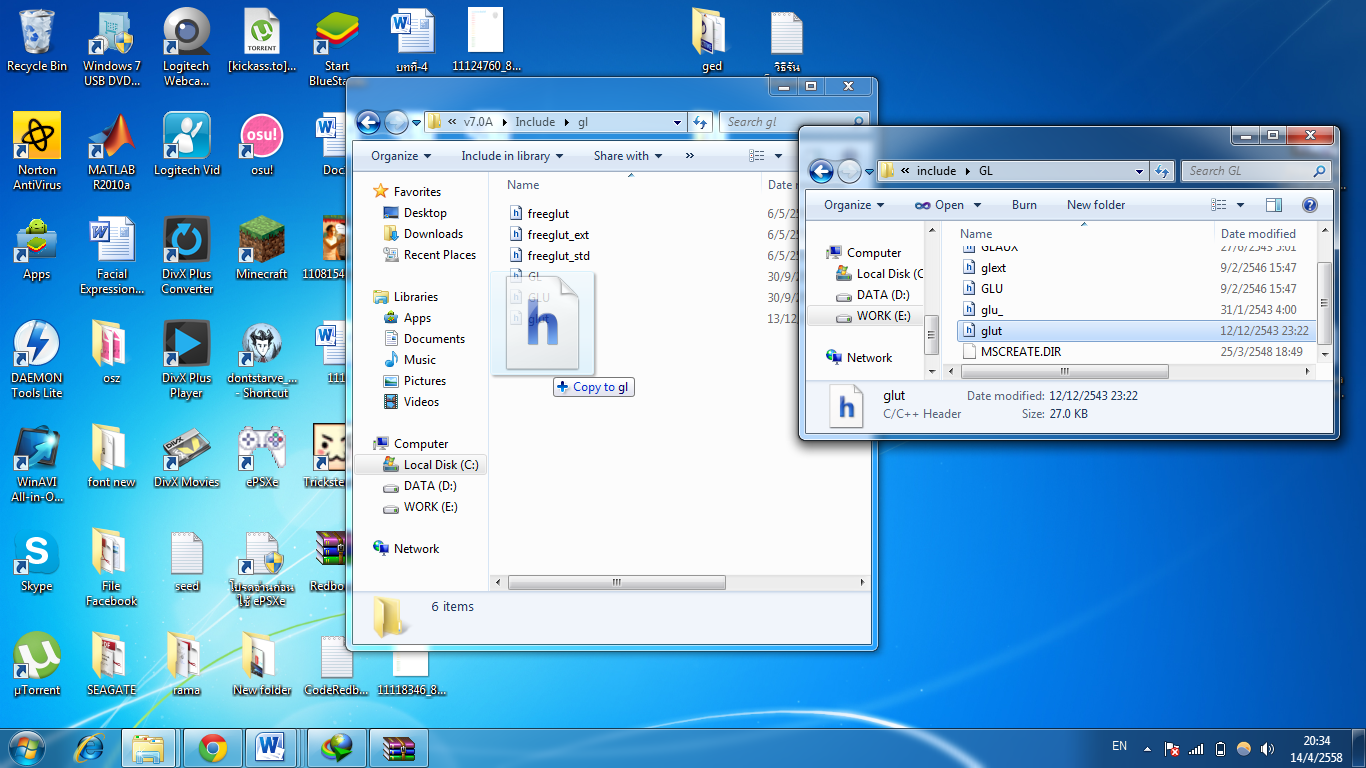
OpenGL คือไลบรารีกราฟิกส์แบบเปิด สามารถใช้สร้างภาพสามมิติได้ โดยในงานวิจัยนี้ผู้วิจัยได้ใช้ไลบาลรี่ OpenGL ร่วมกับโปรแกรม Microsoft Visual Studio C++ เวอร์ชัน 2010 ในWindow7 64bit

**วิธีการติดตั้ง OpenGL**

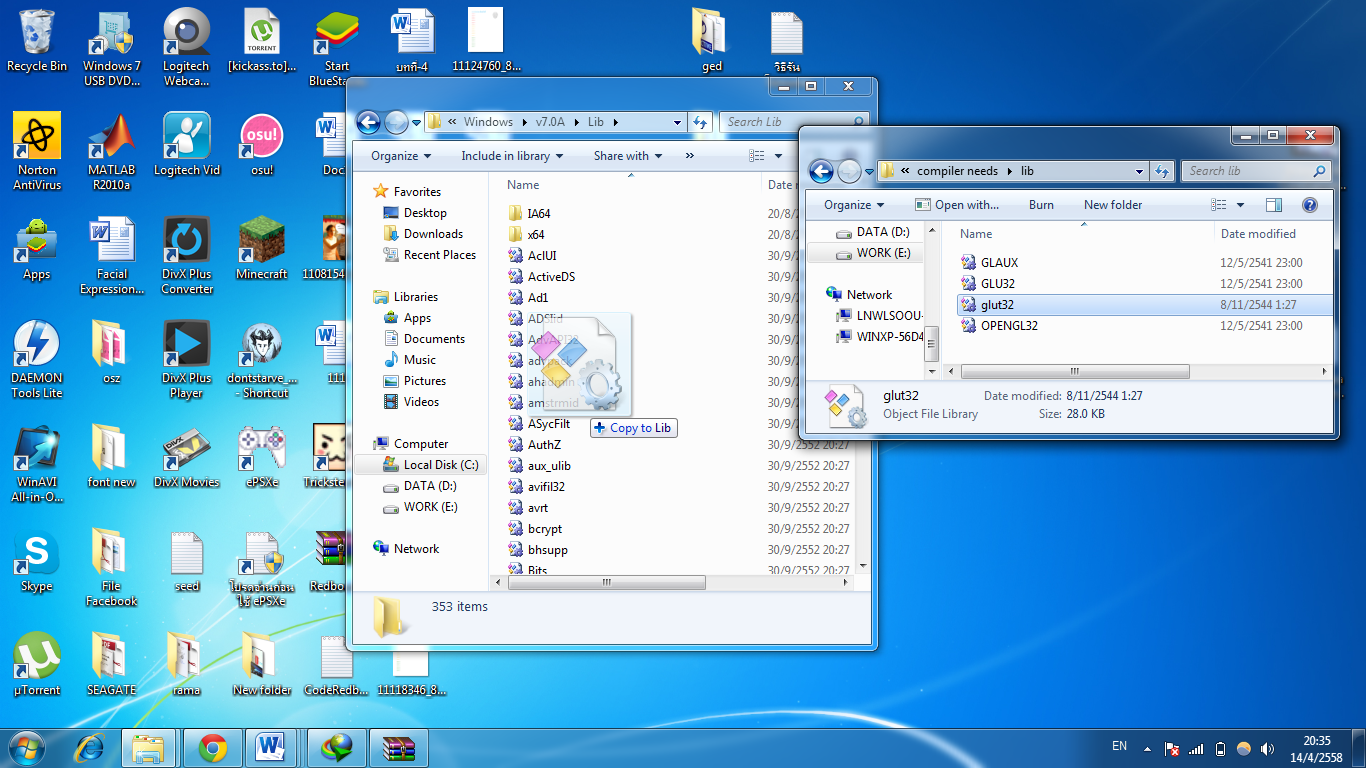
1. ดาวน์โหลดไลบลาลี่ OpenGL จากเว็บไซต์  [opengl.org/resources/libraries/glut](http://www.opengl.org/resources/libraries/glut/) แล้วแตกไฟล์
2. นำไฟล์ glut32.dll ไปไว้ที่ C:\WINDOWS\SysWOW64



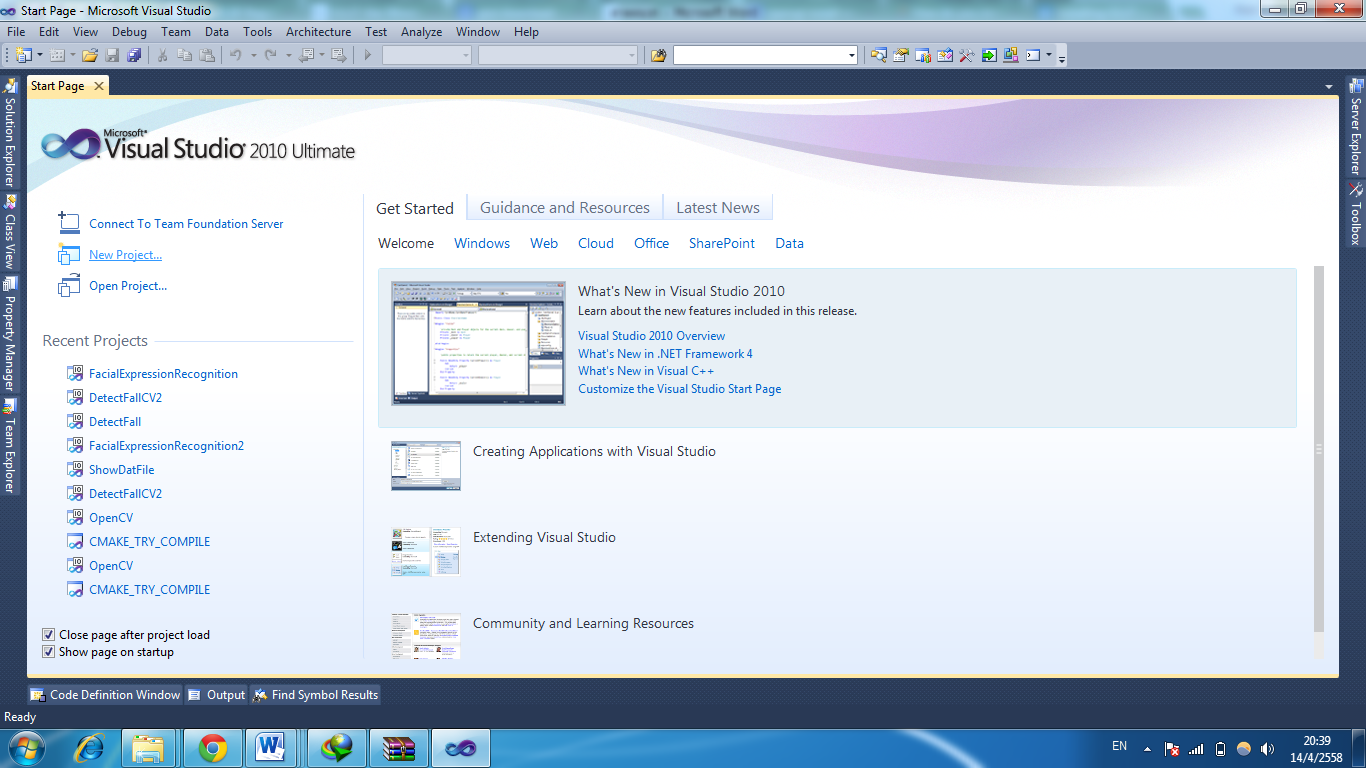
1. ต่อมาให้นำไฟล์ glut.h ไปไว้ที C:\Program Files(x86)\Microsoft SDKs\Windows\v7.0A\include\gl



1. ต่อมาให้ทำการนำไฟล์ glut32.lib ไปไว้ที่ C:\Program Files\Microsoft SDKs\Windows\v7.0A\Lib



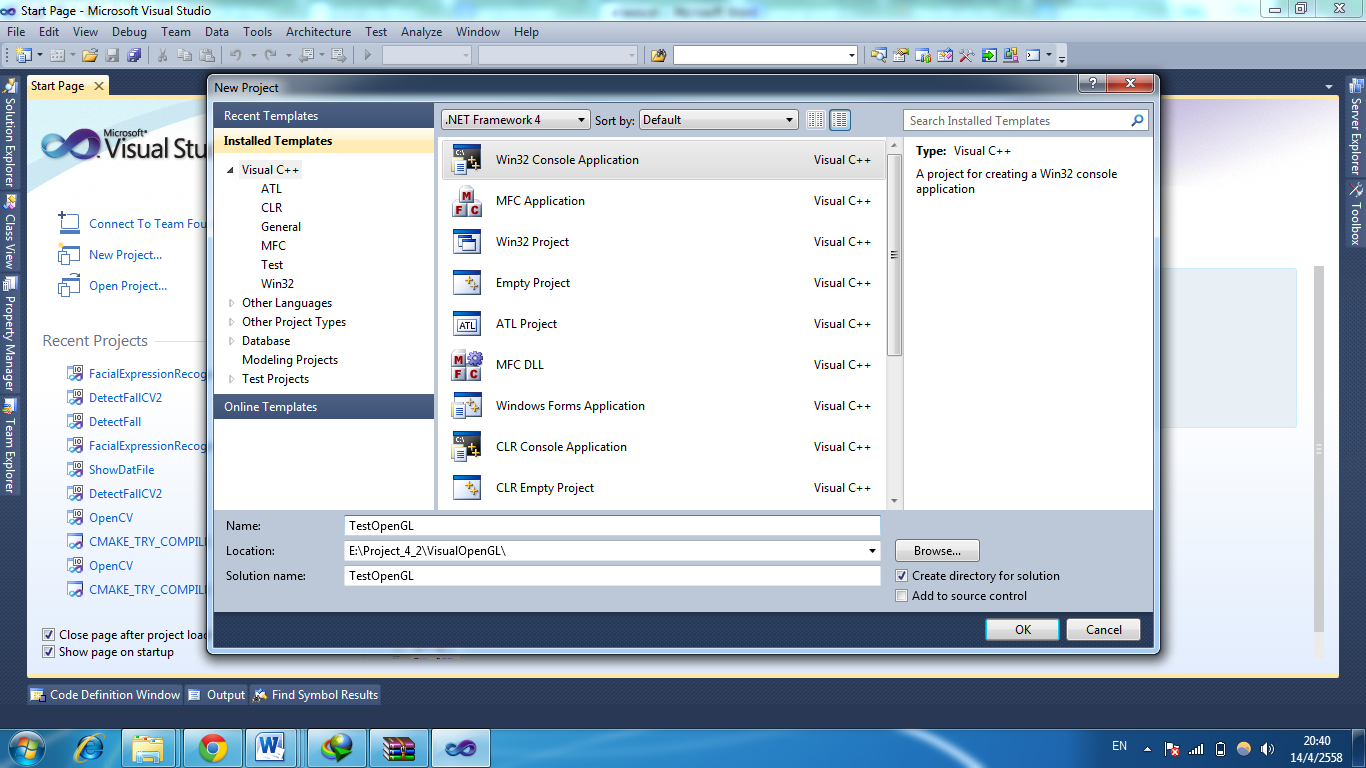
1. เมื่อนำไฟล์ไปที่ต่างๆเรียบร้อยแล้ว ให้ทดลองสร้างโปรเจคใหม่ โดยไปที่โปรแกรม Microsoft Visual Studio 2010 เลือก New Project…



1. ที่ Installed Template เลือก Visual C++
2. เลือก Win32 Console Application
3. ที่ช่อง Name ตั้งชื่อโปรเจค แล้วกำหนดที่จัดเก็บโปรเจคตรงช่อง Location
4. คลิก OK

**6.**

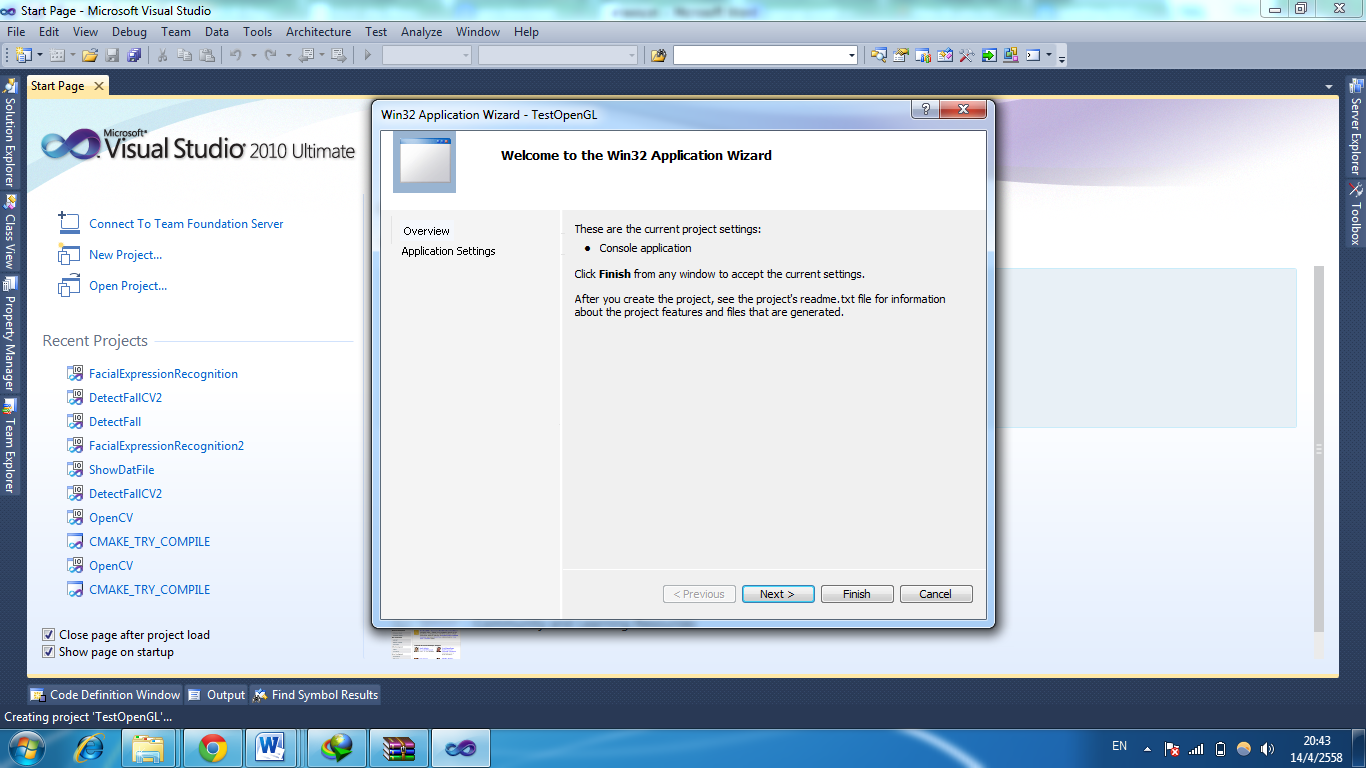
**7.**



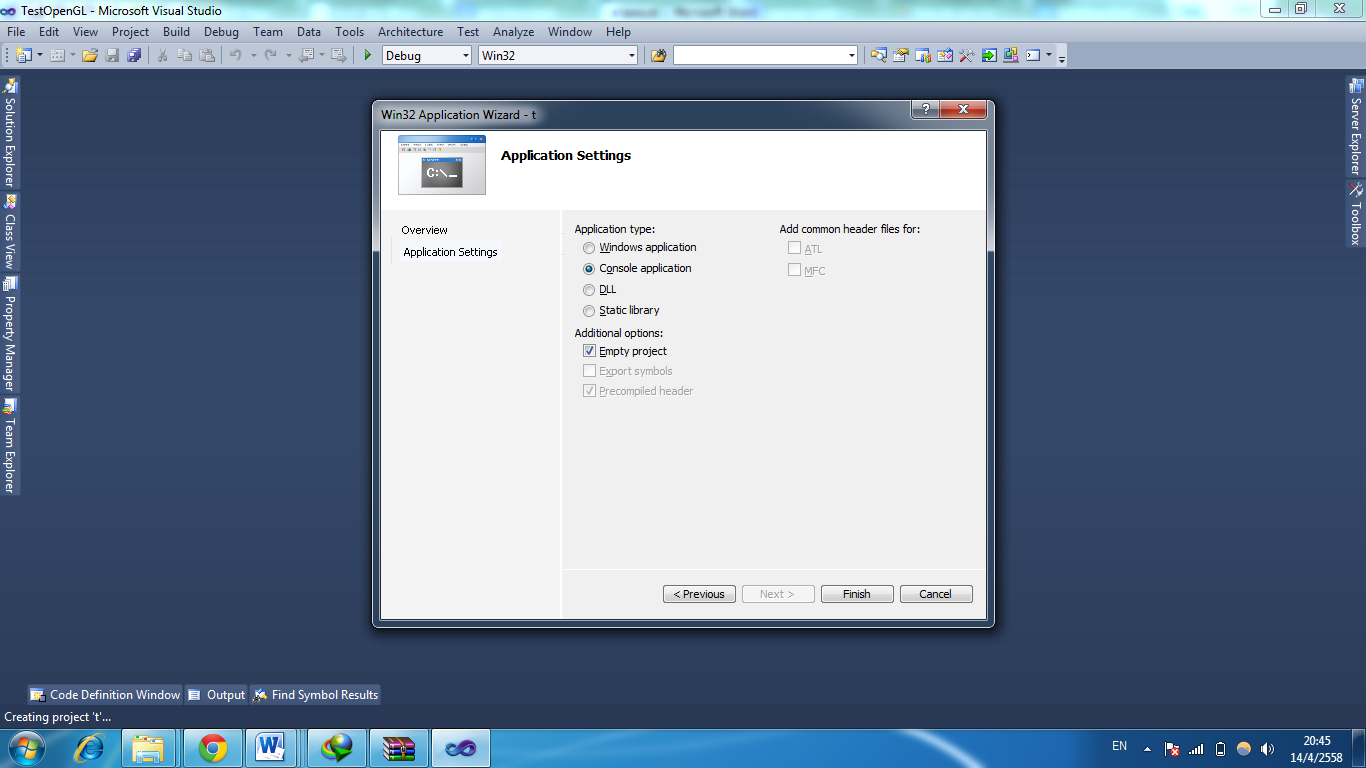
**9.**

**8..**

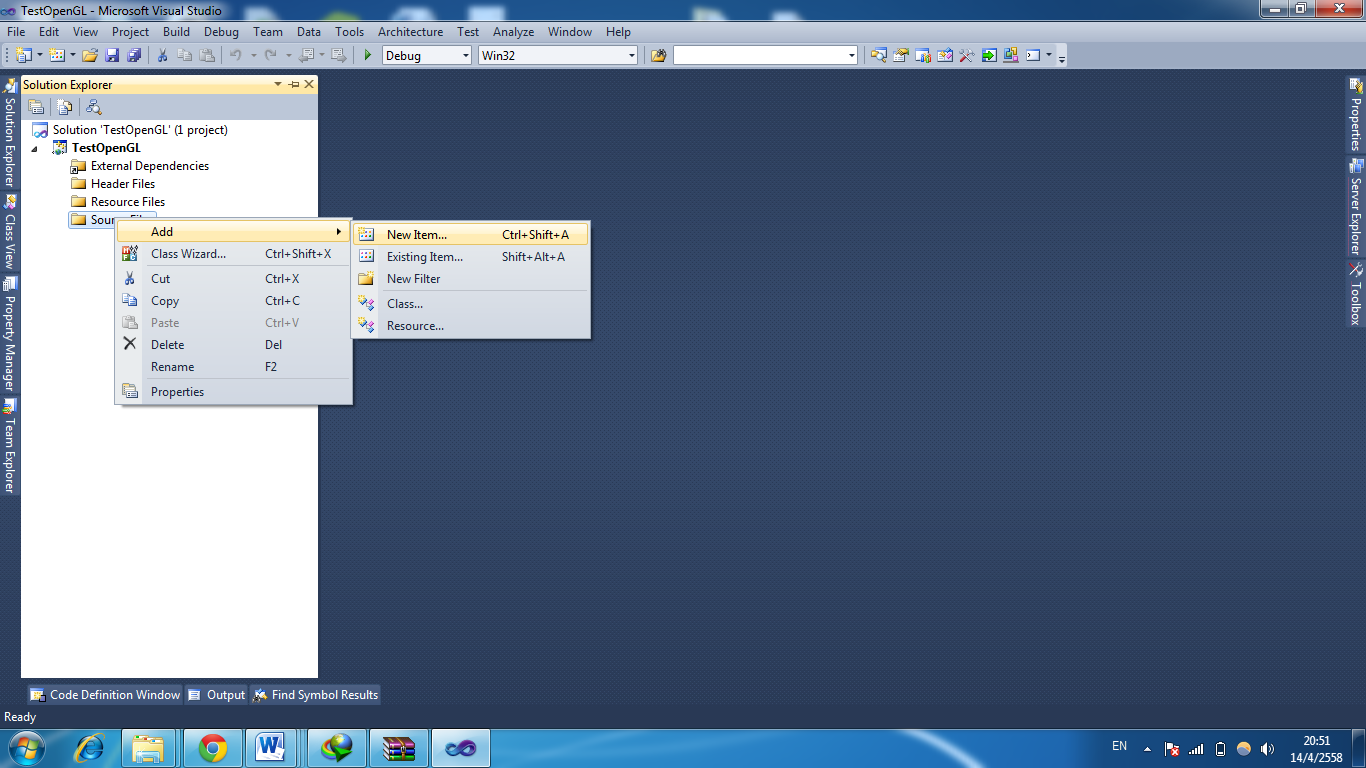
1. คลิก Next



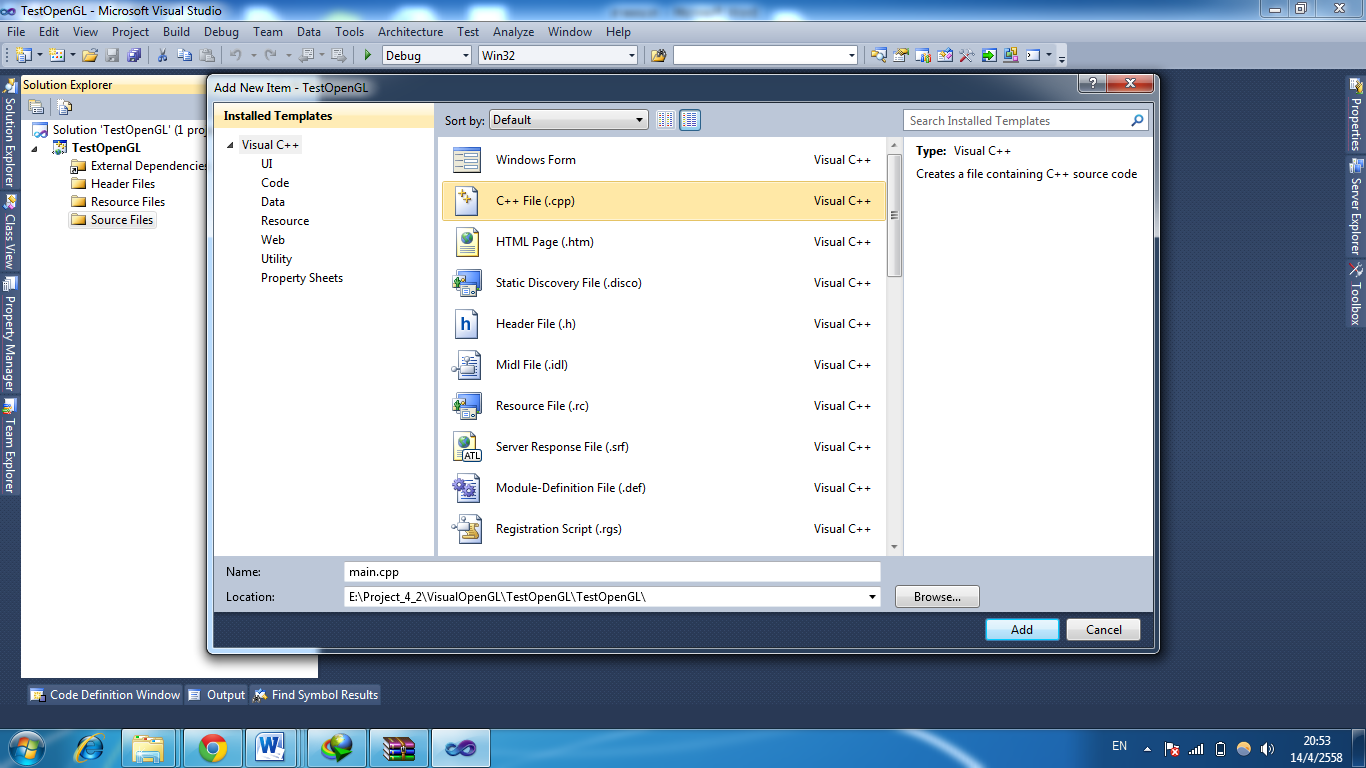
1. ที่ Additional options เลือก Empty project คลิก Finish



1. ใน Solution Explorer คลิกขวาที่ Source Files ไปที่ Add เลือก New Item…



1. เลือก C++ File (.cpp)
2. ที่ช่อง Name ตั้งชื่อว่า main.cpp
3. คลิก Add

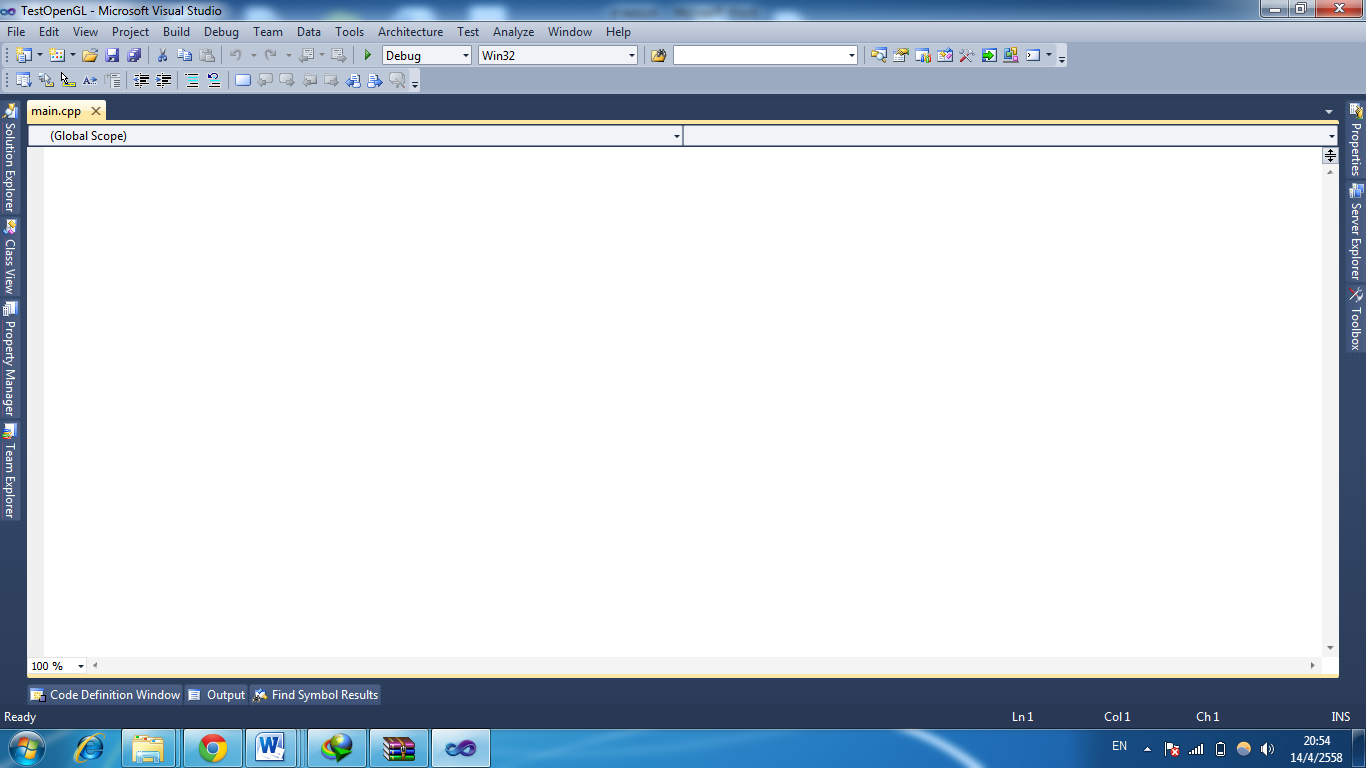


**15.**

**14.**

**13.**

1. จะปรากฏบริเวณให้เขียน code ดังรูป



1. นำโค้ดทดลองไปใส่

#include <windows.h>

#include <GL/gl.h>

#include <GL/glut.h>

int a[3]={10,10,10}, b[3]={10,-10,10},

c[3]={-10,-10,10}, d[3]={-10,10,10},

e[3]={10,10,-10}, f[3]={10,-10,-10},

g[3]={-10,-10,-10}, h[3]={-10,10,-10};

float angle=1.0;

void drawcube(void)

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glColor3f(1.0, 1.0, 1.0);

glMatrixMode(GL\_MODELVIEW);

glRotatef(angle, 0.0, 1.0, 0.0);

glBegin(GL\_LINE\_LOOP);

glVertex3iv(a);

glVertex3iv(b);

glVertex3iv(c);

glVertex3iv(d);

glEnd();

glFlush();

glutSwapBuffers();

}

int main(int argc, char \*\*argv)

{

glutInit(&argc, argv);

glutInitWindowSize(500, 500);

glutInitDisplayMode(GLUT\_RGB | GLUT\_DOUBLE);

glutCreateWindow("Test Opengl");

glutDisplayFunc(drawcube);

glutIdleFunc(drawcube);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

glOrtho(-30.0, 30.0, -30.0, 30.0, -30.0, 30.0);

glRotatef(30.0, 1.0, 0.0, 0.0);

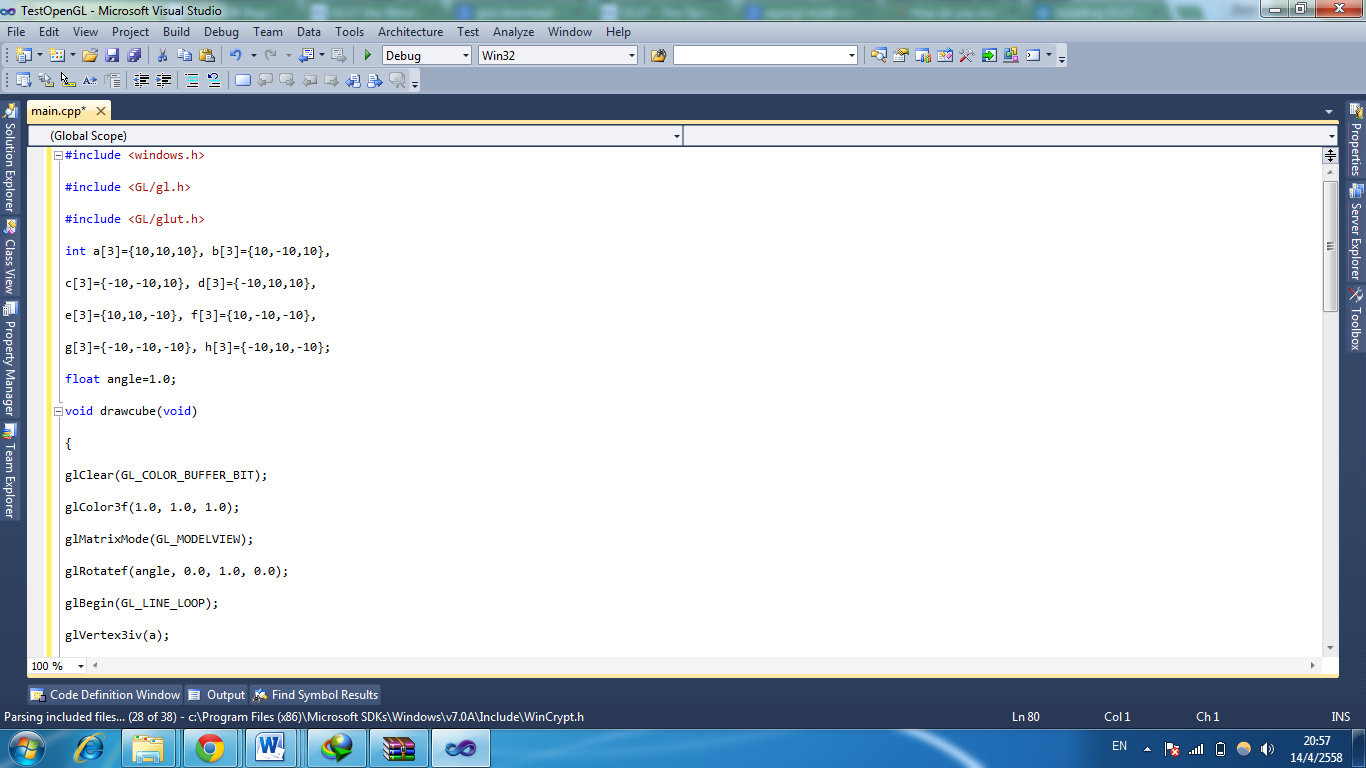
glMatrixMode(GL\_MODELVIEW);

glClearColor(0.0, 0.0, 0.0, 0.0);

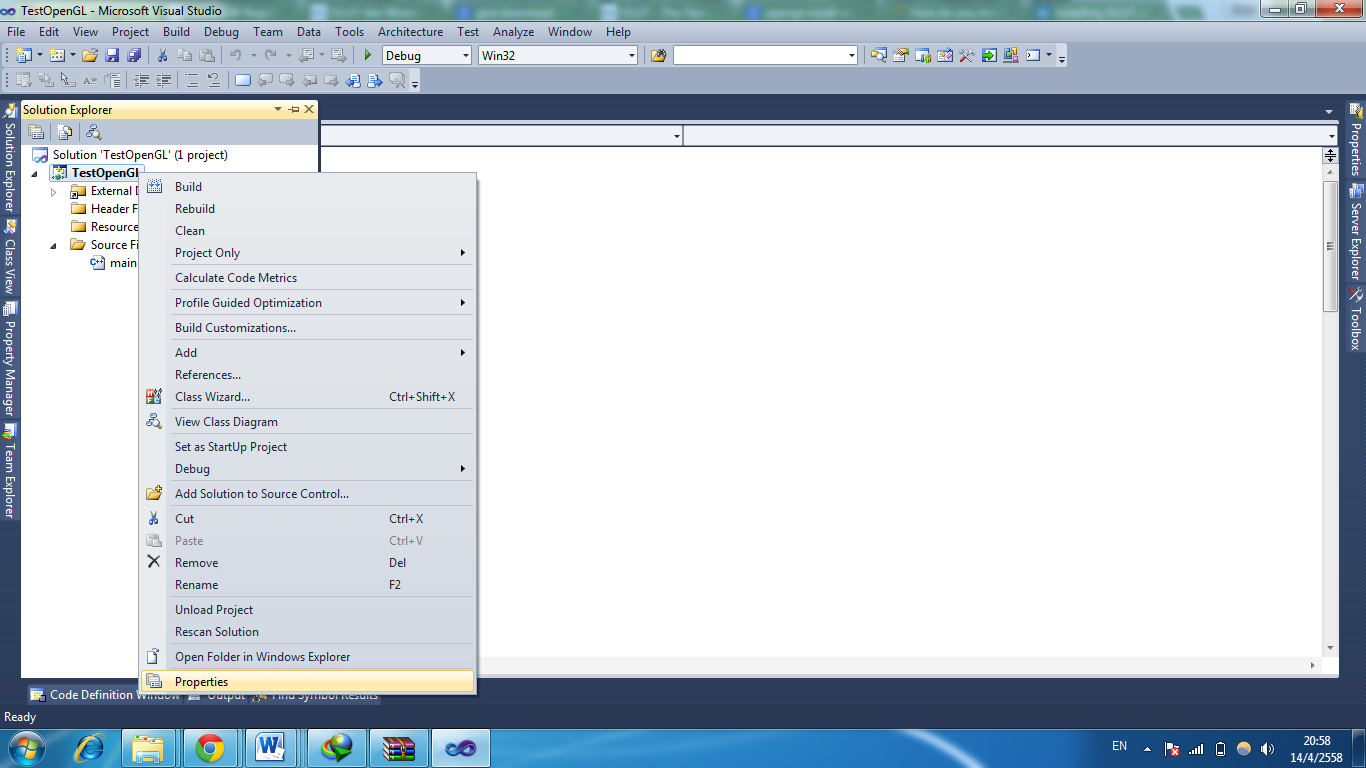
glutMainLoop();

return(0);

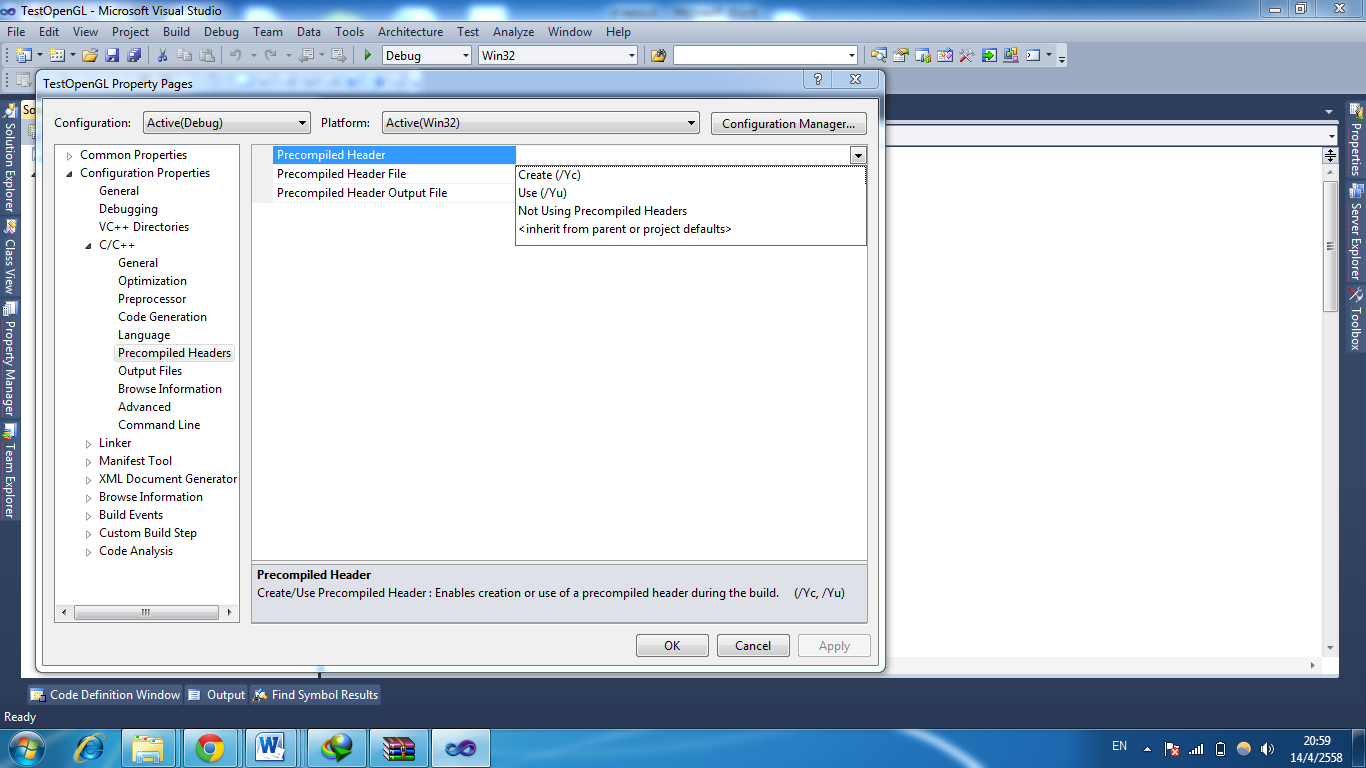
}



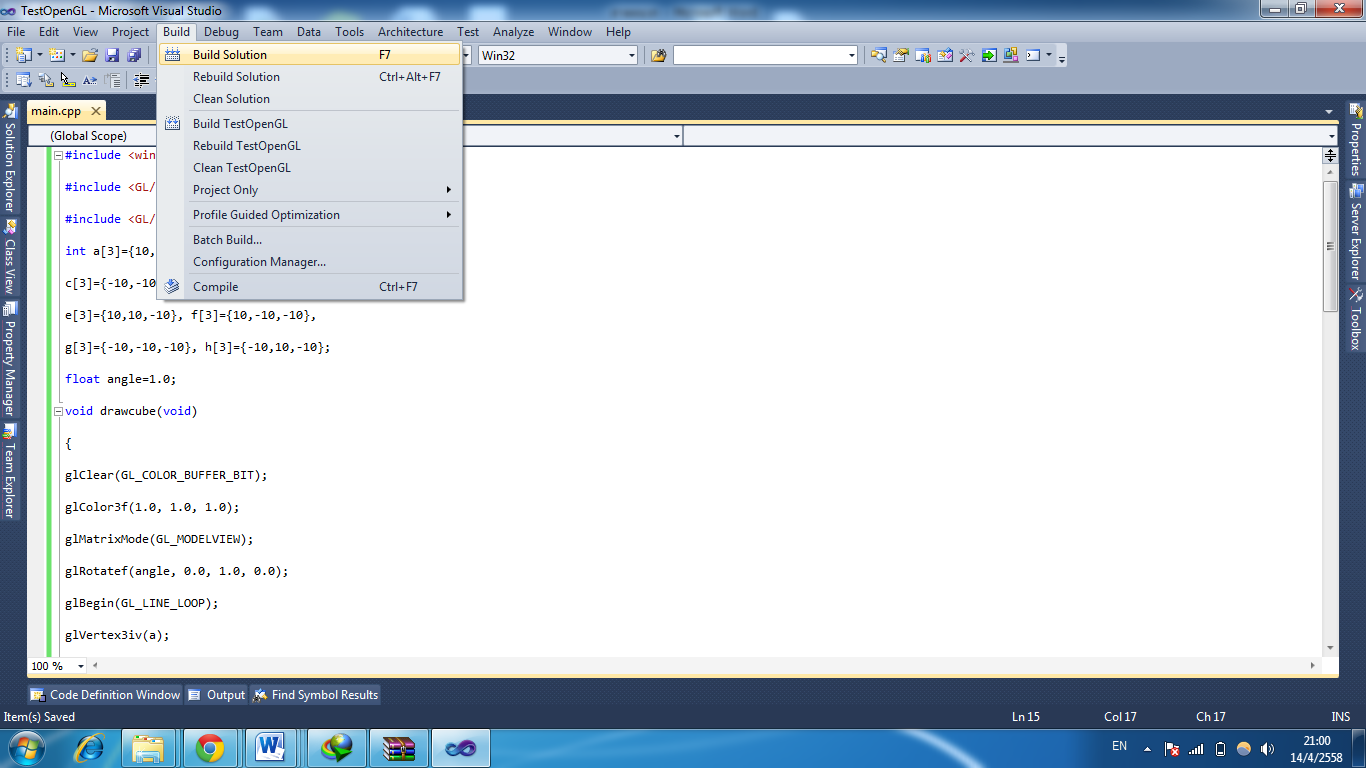
1. คลิกขวาที่ project เลือก properties



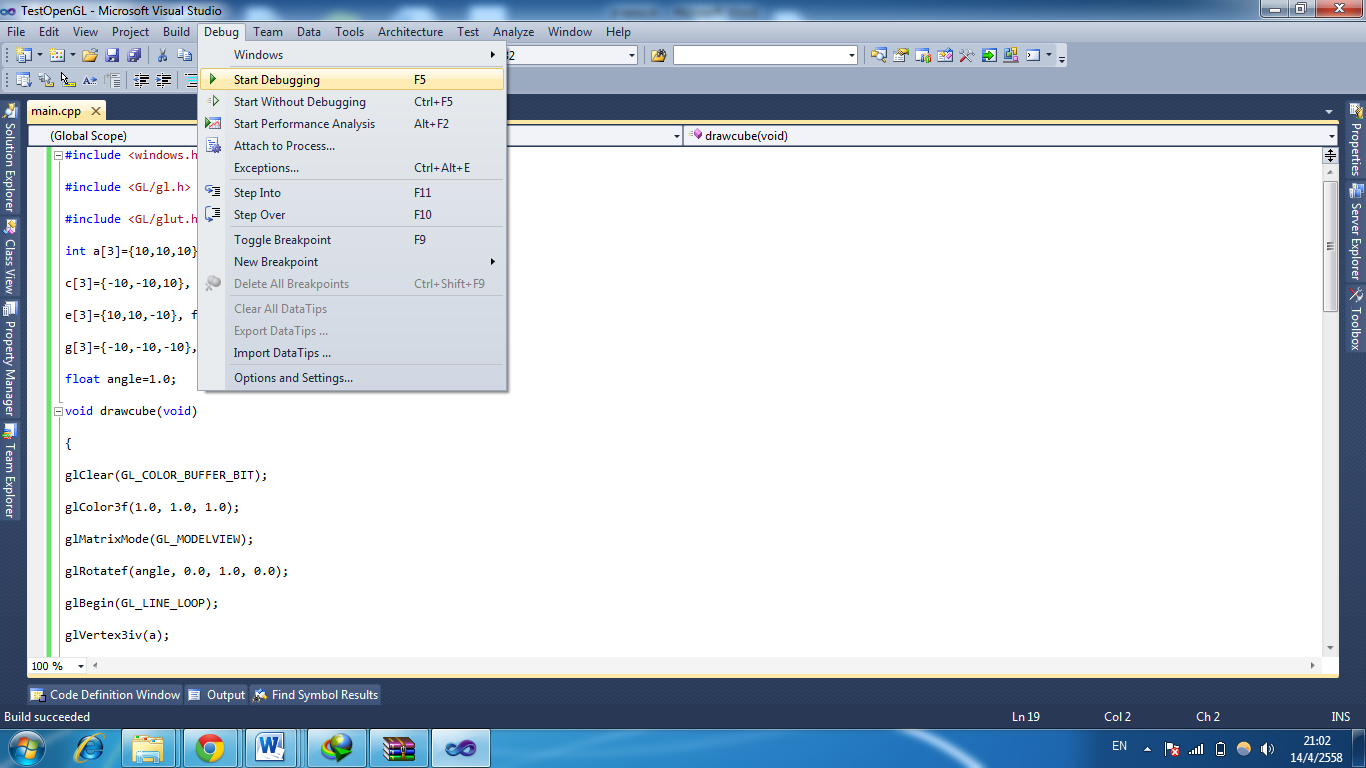
1. จากนั้นเลือก C/C++ แล้วไปที่ Precompiled Headers เลือก Precompiled Header เป็น Not Using Precompiled Headers แล้วคลิก OK



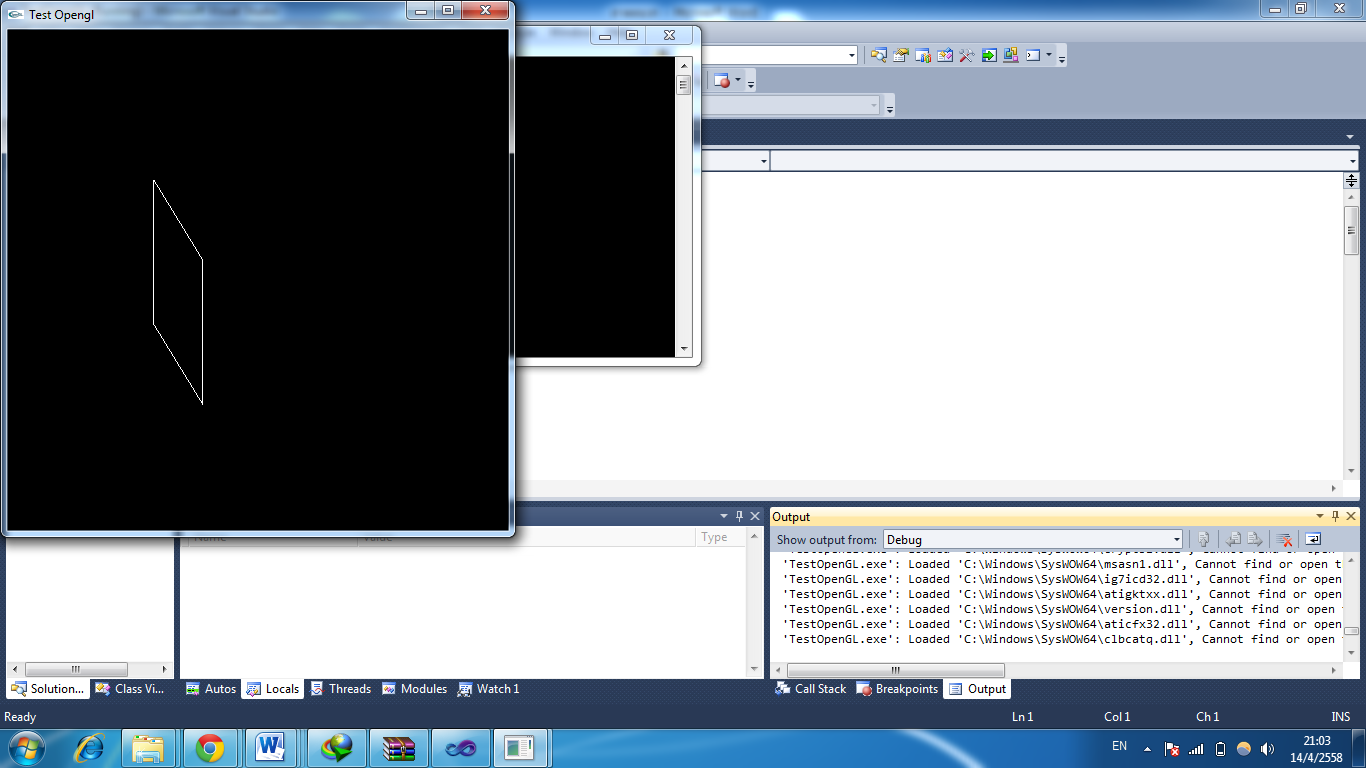
1. Save
2. ไปที่เมนู Build เลือก Build Solution



1. ไปที่เมนู Debug เลือก Start Debugging



1. จะได้ผลลัพธ์ดังรูป



**Source code**

* **การนำเข้าข้อมูลสามมิติจากฐานข้อมูล BU-3DFE**

function [Point, TotPoints] = FaceRead(LocationIn, ImageFile, InputFile)

%LocationIn = 'E:\Project\_4\3D\_database\F0001\';

%InputFile = 'F0001\_NE00WH\_F3D';

%ImageFile = 'F0001\_NE00WH\_F2D';

Txt = fopen([LocationIn,InputFile '.wrl'], 'r');

i = 1;

skp = zeros(1, 100);

while (i<=19)

skp = fgetl(Txt);

i = i+1;

end

i = 1;

Point = zeros(100000, 3);

L0 = 0;

cont = zeros(1, 75);

while (L0 == 0)

cont = fgetl(Txt);

C = ' ]';

L1 = cont(1, 1:41);

if strcmp(L1,C) %(L1 == C)

L0 = 1;

else

Point(i, :) = sscanf(cont, '%f %f %f');

i = i+1;

end

end

TotPoints = i-1;

Point(i:100000,:) = [];

fclose(Txt);

end

* **การวิเคราะห์องค์ประกอบหลัก Principle Component Analysis**

function [pca]=PCA(DAll,numEmo)

%[DAll] = ReadTxtForPCA;

% DAll คือเวกเตอร์แสดงจำนวนจุดเฉลี่ยในแต่ละช่อง มีขนาด 14\*14\*numEmo

% numEmo คือจำนวนอารมณ์

Size = size(DAll);

Mean = mean(mean(DAll),3);

SigmaCo = zeros(Size(1,2),Size(1,2));

pca = zeros(196,1);

for j=1:numEmo

for i=1:Size(1,1)

S=(DAll(i,:,j)'-Mean')\*(DAll(i,:,j)'-Mean')';

SigmaCo = SigmaCo + S;

end

end

Covariance = SigmaCo/(Size(1,1)\*Size(1,3));

C\_Diag = diag(Covariance);

C\_Diag2 = diag(Covariance);

for i = 1:196

[C,I] = max(C\_Diag2);

C\_Diag2(I,1) = 0;

pca(i)=I;

end

end

* **การจำแนกข้อมูลด้วยเทคนิค Support Vector Machine**

function [result] = multisvm(TrainingSet,GroupTrain,TestSet)

%Models a given training set with a corresponding group vector and

%classifies a given test set using an SVM classifier according to a

%one vs. all relation.

%

%This code was written by Cody Neuburger cneuburg@fau.edu

%Florida Atlantic University, Florida USA

%This code was adapted and cleaned from Anand Mishra's multisvm function

%found at http://www.mathworks.com/matlabcentral/fileexchange/33170-multi-class-support-vector-machine/

u=unique(GroupTrain);

numClasses=length(u);

result = zeros(length(TestSet(:,1)),1);

%build models

for k=1:numClasses

%Vectorized statement that binarizes Group

%where 1 is the current class and 0 is all other classes

G1vAll=(GroupTrain==u(k));

models(k) = svmtrain(TrainingSet,G1vAll);

end

%classify test cases

for j=1:size(TestSet,1)

for k=1:numClasses

if(svmclassify(models(k),TestSet(j,:)))

break;

end

end

result(j) = k;

end