

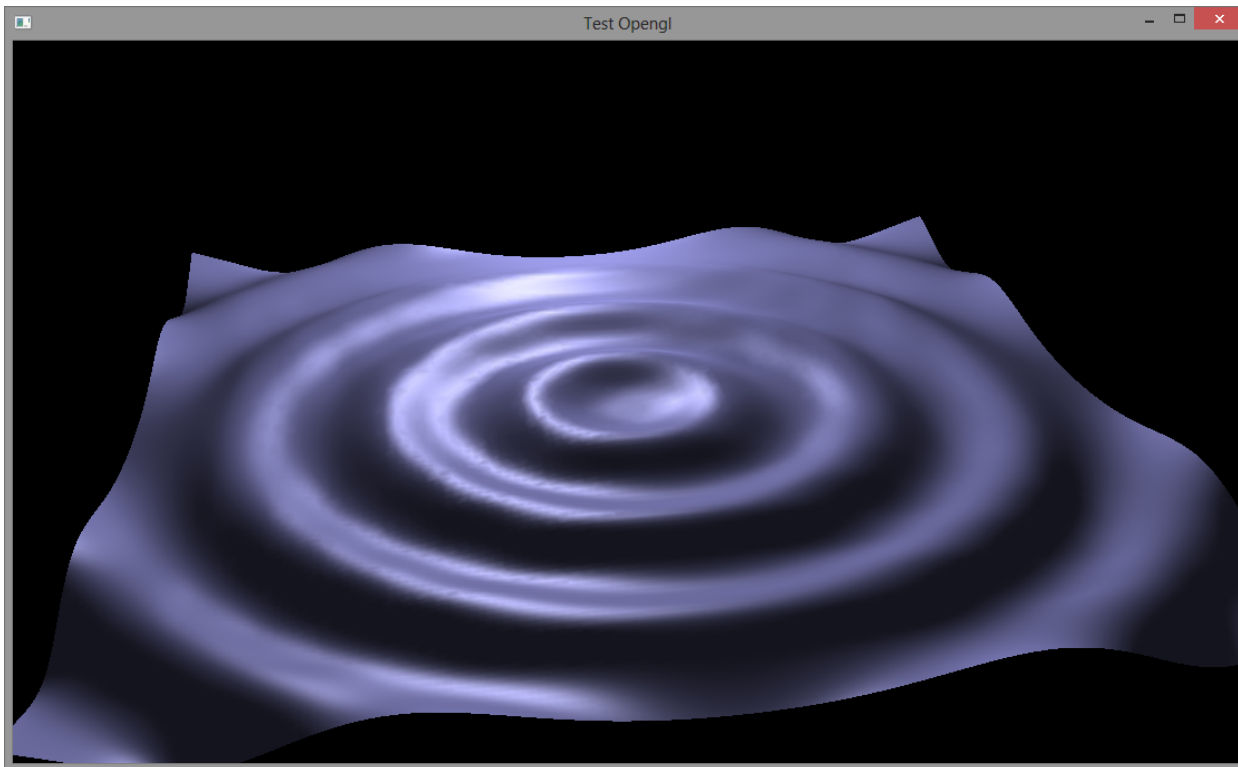
## Proyecto III - Opengl

Usando la siguiente referencia:

[http://http.developer.nvidia.com/GPUGems/gpugems\\_ch01.html](http://http.developer.nvidia.com/GPUGems/gpugems_ch01.html)

Deberán de crear una ola circular, para ello deben utilizar la ecuación (equation 1) con la adaptación del  $D_i$  que se encuentra en la sección *Directional or Circular*.

Dicha Ola deberá animarse, sin embargo eso está contemplado en la ecuación en la variable  $t$ .



Ademas de crear la ola, deberán extraer de la siguiente página el código para crear un efecto de ruido:

<http://mrl.nyu.edu/~perlin/doc/oscar.html>

Deben utilizar el noise2, el cual recibe un vector de 2 coordenadas, dicho vector deberá seguir la siguiente fórmula:

```
n_noise[0] = coord_original_punto_control_x*Amplitud_Ruido + Offset_ruido ;  
n_noise[1] = coord_original_punto_control_z*Amplitud_Ruido + Offset_ruido ;
```

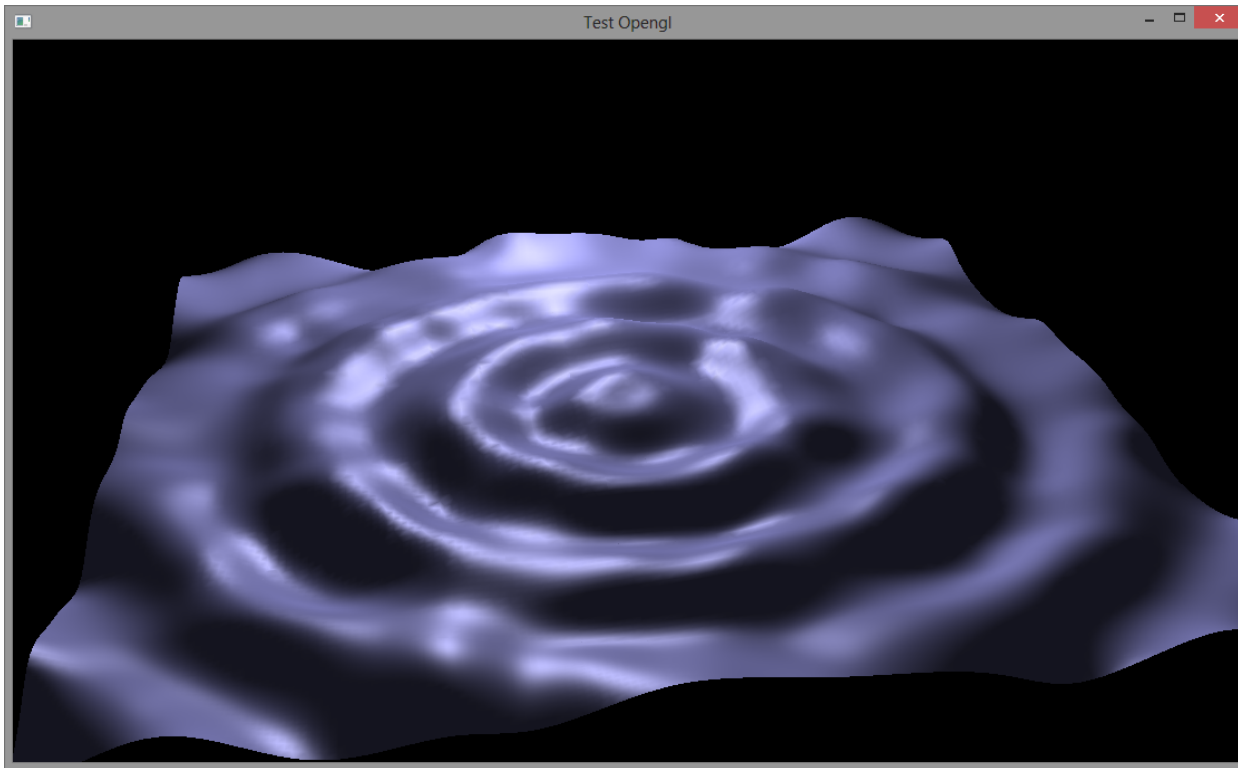
Luego de este link:

<http://lodev.org/cgtutor/randomnoise.html#Turbulence>

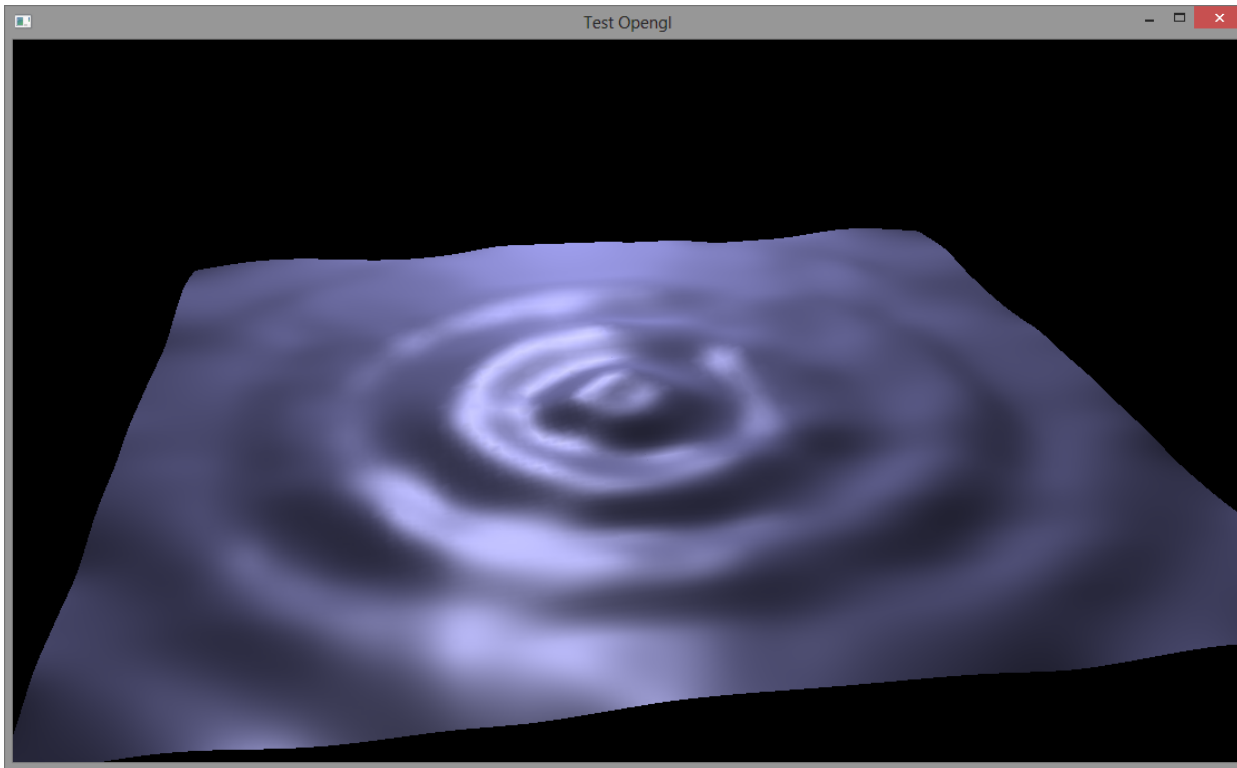
Toman la función turbulence, el X e Y son las dos coordenadas previas, size será un parámetro de entrada por el usuario. Usan la función noise2 en sustitución de la función smoothnoise.

```
ruido = Altura_Ruido * 0.005 * turbulence(n_noise[0], n_noise[1], factorTurb)
```

Usen el valor para romper la perfección de las olas:

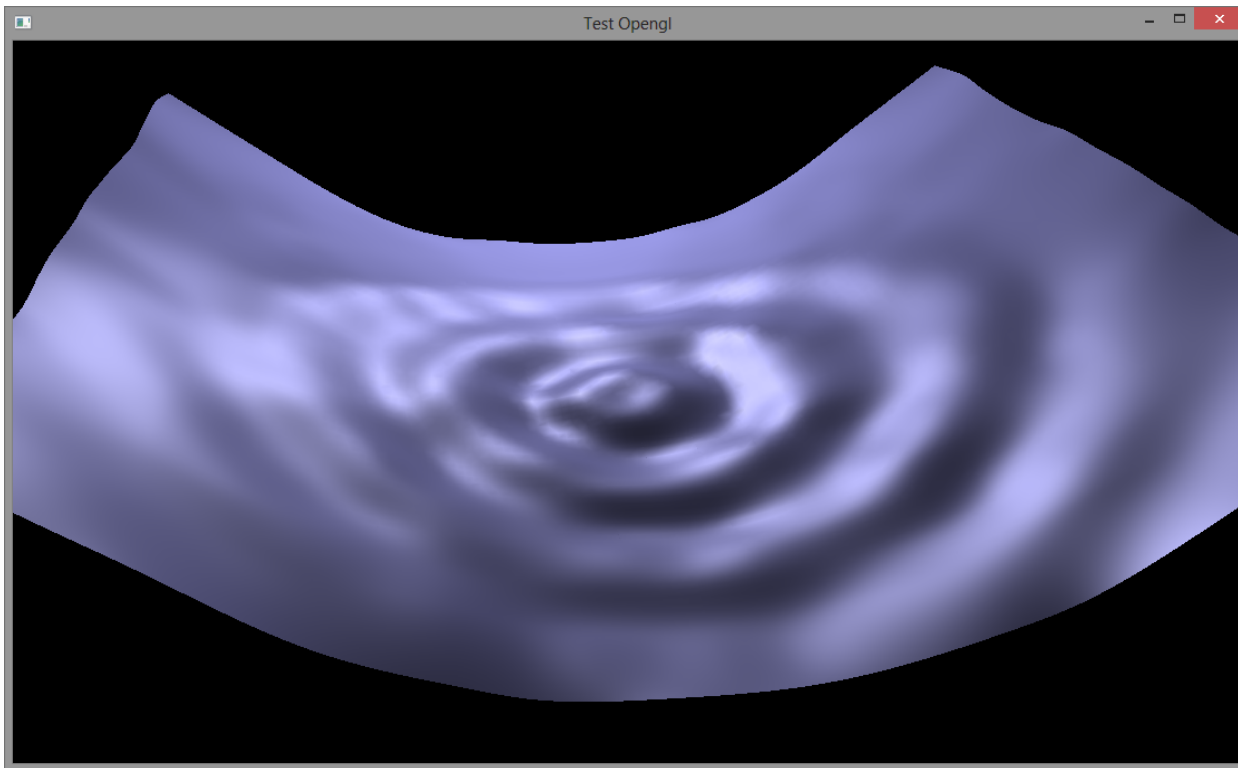


Como las olas no mantienen la misma altura todo el tiempo, deberán de crear un decaimiento, el factor de decaimiento lo especifica el usuario.



Por último, deberán de poder doblar el plano de las olas en forma cuadrática, y se debe poder trasladar el centro de dicha parábola.

**NOTA:** con el proyecto se les pasa una plantilla (`mainNurbsProyecto_base01.cpp`) en el render ya se encuentra la función en comentarios que dibuja la superficie nurbs, deberán usar 21x21 puntos de control, y tendrán que crear la matriz tridimensional con los puntos de control (ver última imagen del documento), y el vector de Knots. Recuerden que la altura es el eje Y.



Estos dos últimos puntos, deberán resolverlo con lo visto en clase.

A continuación muestro las teclas que deberá usar el usuario para cambiar los distintos valores. Teclas:

- a = aumenta en 0.1 la amplitud de la ola.
- z = disminuye en 0.1 la amplitud de la ola.
- s = aumenta en 0.1 la longitud de la ola.
- x = disminuye en 0.1 la longitud de la ola.
- d = aumenta en 0.1 la velocidad de la ola.
- c = disminuye en 0.1 la velocidad de la ola.
- f = aumenta en 0.01 el decaimiento de la ola.

v = disminuye en 0.01 el decaimiento de la ola.  
g = aumenta en 0.01 la amplitud del ruido.  
b = disminuye en 0.01 la amplitud del ruido.  
h = aumenta en 0.01 el offset del ruido.  
n = disminuye en 0.01 el offset del ruido.  
j = aumenta en 0.01 la altura del ruido.  
m = disminuye en 0.01 la altura del ruido.  
t = aumenta en 1 el size de la función turbulencia.  
y = disminuye en 1 el size de la función turbulencia.  
u = aumenta en 0.001 la amplitud de la curva cuadrática (deformador).  
i = disminuye en 0.001 la amplitud de la curva cuadrática (deformador).  
o = aumenta en 0.1 la traslación en X de la curva cuadrática.  
l = disminuye en 0.1 la traslación en X de la curva cuadrática.  
q = aumenta en 0.1 el centro en el eje X de la ola.  
w = disminuye en 0.1 el centro en el eje X de la ola.  
e = aumenta en 0.1 el centro en el eje Z de la ola.  
r = disminuye en 0.1 el centro en el eje Z de la ola.  
1 = pone en pausa la animación.  
2 = desactiva el ruido.  
3 = desactiva la ola.

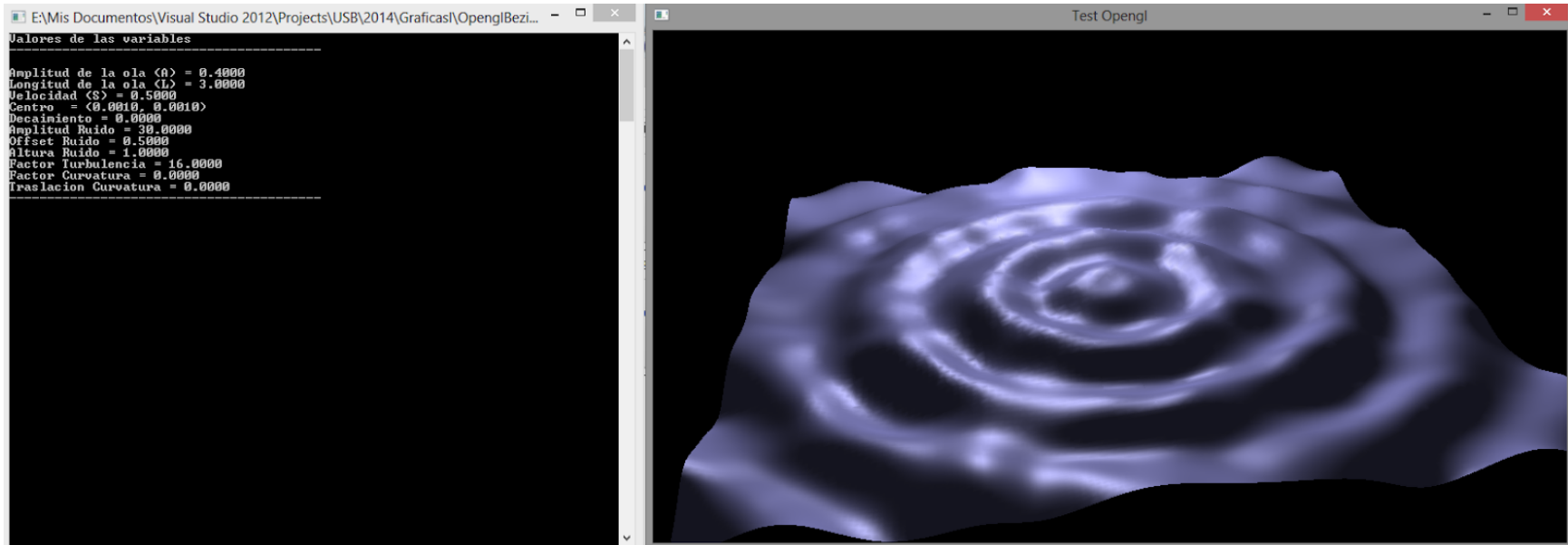
### Especificaciones del proyecto:

1. El proyecto se puede crear en **parejas**.
2. El proyecto tiene un **15%** de nota.
3. La entrega es el **sabado 07/02**.
4. Deberán enviarme por correo (comprimido en .rar) el código fuente (solo los archivos .cpp, no es necesario comprimir todo el proyecto) y el ejecutable.

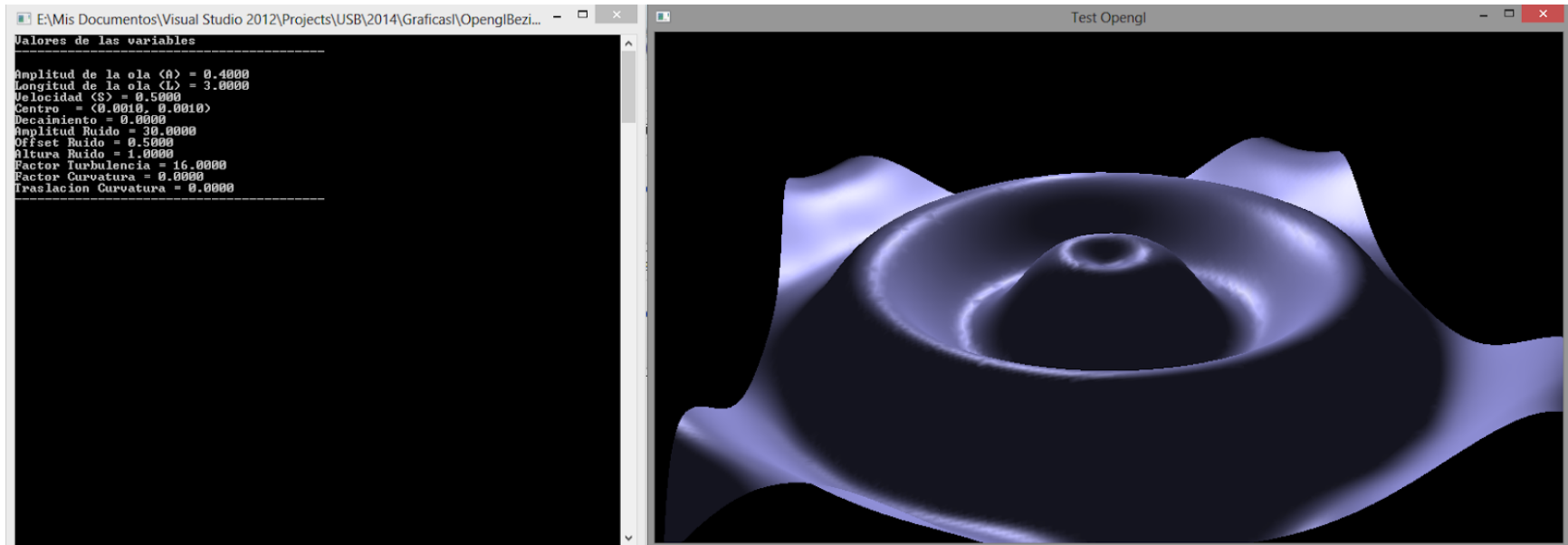
IMPORTANTE: el asunto del correo debe estar escrito con el siguiente formato: **"Graficas I 2015 - Proyecto III - Opengl - NombresEstudiantes"**.

Ejemplos:

ruido = true, ola = true

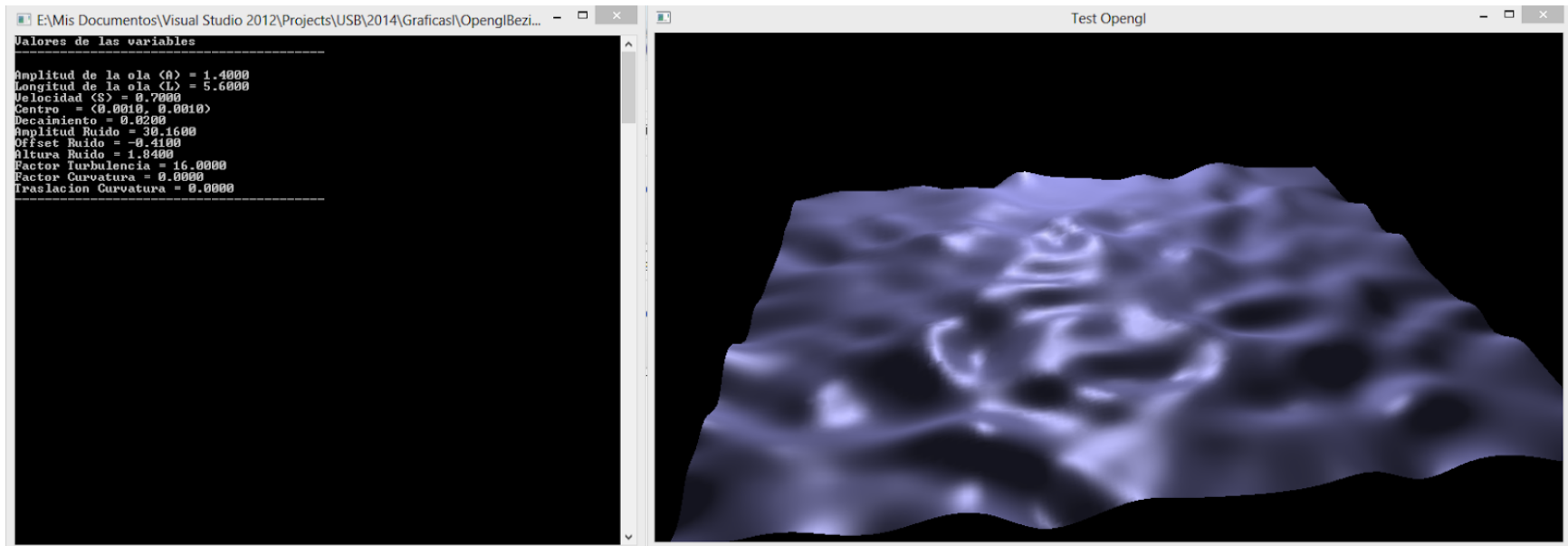


ruido = false, ola = true

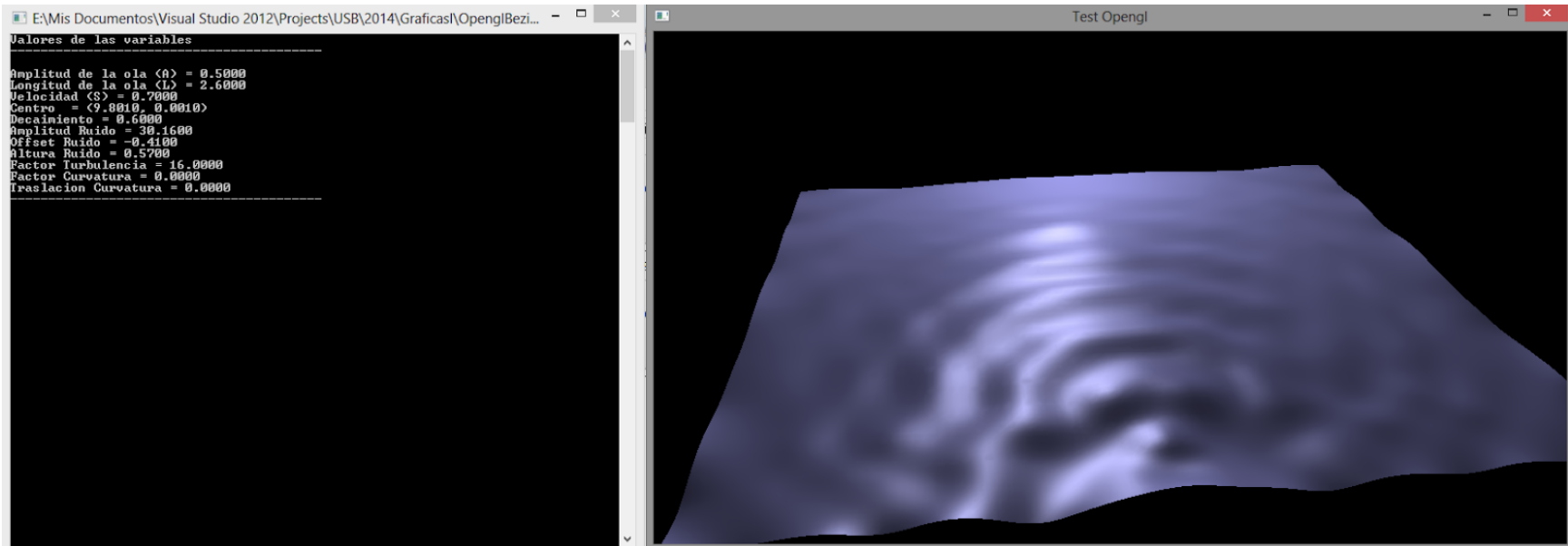




ruido = true, ola = false



ruido = true, ola = true



ruido = true, ola = false.

