Simplex Noise Documentation

This extension provides an easy way to calculate 2d, 3d, and 4d simplex noise. The scripts are available for free on the GMC forums, while this extension is a more refined compilation of those scripts.

simplex_set_seed(seed)

Description: *Sets a new seed for the algorithm and regenerates the necessary hash values.*
Argument0: *(int) the seed to use*
Returns: *(real) seed specified*

simplex_get_seed()

Description: *Returns the current seed being used by the simplex noise algorithms.*
Returns: *(real) current seed*

simplex_calculate_2d(x, y, min, max, octaves, persistence, scale)

Description: *Generates fractal 2D simplex noise for a single coordinate. The result is scaled between [min .. max].*
Argument0: *(double) x-pos to calculate*
Argument1: *(double) y-pos to calculate*
Argument2: *(double) lower-end of the scale*
Argument3: *(double) higher-end of the scale*
Argument4: *(double) number of octaves to sample over*
Argument5: *(double) change of octave intensity*
Argument6: *(double) scale of octave intensity delta*
Returns: *(double) resulting value between [min .. max]*

simplex_calculate_3d(x, y, min, max, octaves, persistence, scale)

Description: *Generates fractal 3D simplex noise for a single coordinate. The result is scaled between [min .. max].*
Argument0: *(double) x-pos to calculate*
Argument1: *(double) y-pos to calculate*
Argument2: *(double) z-pos to calculate*
Argument3: *(double) lower-end of the scale*
Argument4: *(double) higher-end of the scale*
Argument5: *(double) number of octaves to sample over*
Argument6: *(double) change of octave intensity*
Argument7: *(double) scale of octave intensity delta*
Returns: *(double) resulting value between [min .. max]*

simplex_calculate_4d(x, y, z, w, min, max, octaves, persistence, scale)

Description: Generates fractal 2D simplex noise for a single coordinate. The result is scaled between [min .. max].

Argument0: (double) x-pos to calculate

Argument1: (double) y-pos to calculate

Argument2: (double) z-pos to calculate

Argument3: (double) w-pos to calculate

Argument4: (double) lower-end of the scale

Argument5: (double) higher-end of the scale

Argument6: (double) number of octaves to sample over

Argument7: (double) change of octave intensity

Argument8: (double) scale of octave intensity delta

Returns: (double) resulting value between [min .. max]

simplex_calculate_2dr(x, y, min, max, octaves, persistence, scale)

Description: Generates ridged fractal 2D simplex noise for a single coordinate. The result is scaled between [min .. max].

Argument0: (double) x-pos to calculate

Argument1: (double) y-pos to calculate

Argument2: (double) lower-end of the scale

Argument3: (double) higher-end of the scale

Argument4: (double) number of octaves to sample over

Argument5: (double) change of octave intensity

Argument6: (double) scale of octave intensity delta

Returns: (double) resulting value between [min .. max]

simplex_calculate_3dr(x, y, min, max, octaves, persistence, scale)

Description: Generates ridged fractal 3D simplex noise for a single coordinate. The result is scaled between [min .. max].

Argument0: (double) x-pos to calculate

Argument1: (double) y-pos to calculate

Argument2: (double) z-pos to calculate

Argument3: (double) lower-end of the scale

Argument4: (double) higher-end of the scale

Argument5: (double) number of octaves to sample over

Argument6: (double) change of octave intensity

Argument7: (double) scale of octave intensity delta

Returns: (double) resulting value between [min .. max] *simplex_calculate_4dr(x, y, z, w, min, max, octaves, persistence, scale)*

Description: Generates ridged fractal 2D simplex noise for a single coordinate. The result is scaled between [min .. max].

Argument0: (double) x-pos to calculate

Argument1: (double) y-pos to calculate

Argument2: (double) z-pos to calculate

Argument3: (double) w-pos to calculate

Argument4: (double) lower-end of the scale

Argument5: (double) higher-end of the scale

Argument6: (double) number of octaves to sample over

Argument7: (double) change of octave intensity

Argument8: (double) scale of octave intensity delta

Returns: (double) resulting value between [min .. max]