



## CURSO DE ARTE GENERATIVO

Aprender programación es un arte

### RELACIONATE CON EL ESPACIO, LAS FORMAS Y EL MOVIMIENTO A TRAVEZ DE LA PROGRAMACIÓN

#### • | Convoca

Thinking Dojo - Escuela de Arte y Diseño Interactivo

#### • | Objetivo

Al finalizar este curso tendrás herramientas para transformar tus ideas artísticas en instrucciones para una computadora. Te desenvolverás con fluidez dentro de espacios 2d y 3d, y generarás principios sólidos de programación funcional.

#### • | Información General

El espacio, los puntos, las rectas, los figuras y los cuerpos. Todo tiene una descripción, todo se puede definir, de forma precisa, de forma exacta. ¿Y para qué? Pues para que una computadora pueda jugar a ser el asistente del pintor, el ayudante del arquitecto, el dibujante aprendiz. Déjate sorprender por lo que le puedes enseñar, por lo que juntos harán.



#### • | ¿Es para ti?

##### Perfil del alumno

Este curso comienza desde los fundamentos por lo que no necesitas tener conocimientos previos. Para sacarle el mayor provecho te recomendamos tener nociones de diseño o de arte y una buena apreciación estética.

#### • | Superpoderes que tendrás

##### Aprendizajes esperados

- ° Creación de geometrías generativas
- ° Pasar ideas gráficas a algoritmos
- ° Manipulación del espacio 2D y 3D
- ° Comprensión de la mecánica del espacio
- ° Diseño de algoritmos geométricos
- ° Diseño de flujos de información y datos

Para apartar tu lugar es necesario que realices tu pago una semana antes de comenzar el curso.

- | Requisitos para la entrega del diploma

Es necesario pasar satisfactoriamente la evaluación final para poder obtener el certificado del curso. El alumno que haya participado el 90% de las clases, tendrá derecho a una constancia de asistencia.



- | Duración

3 a 4 meses de 3 a 4 hrs.  
1 ó 2 clases por semana

- | Inversión

Individual

Inscripción \$650  
Mensualidad \$1,600

Estudiantes o Grupos

Inscripción \$500  
Mensualidad \$1,300

- | Iniciamos

8 y 13 de junio de 2015

## TEMARIO

- | Bloque Básico

1. ¿Qué son, para qué sirve y cómo se usan los lenguajes de programación?

Primitivas y palabras reservadas

Operadores básicos y sus jerarquías

2. Sistemas de representación y de coordenadas

Métodos de dibujo direccionales

Representaciones sobre el plano

3. Procedimientos, algoritmos y repeticiones

Loops de programación

Procedimientos con parámetros

4. Espacios en memoria, las variables y los parámetros

Procedimientos con parámetros

La variable como parámetro asignado

5. Modularidad, partiendo un problema complejo en varios simples

Planeación de procedimientos modulares

Síntesis deductiva

- | Bloque Intermedio

1. Formación de patrones

Elementos que construyen patrones geométricos

Construcción de figuras complejas a través de la identificación de patrones

2. Razones y proporciones

Uso profundo de variables  
Reducciones y aplicaciones, proporción

3. Relaciones funcionales (Pre-partículas)

Manejo del color en la programación

Introducción al concepto de relación funcional

Fuentes de error en cálculos

4. Recursividad

Pensamiento recursivo

Iteraciones de cola

Iteraciones complejas

Transformaciones espaciales

Espirales

## • | Bloque Avanzado

1. Formación de patrones 3D
  - Elementos básicos
  - Transformaciones
  - Funciones aleatorias
  - Experimentación
  - Relaciones funcionales
2. Animaciones
  - Sistema de refresco de pantalla
  - Animaciones mediante loops
  - Animaciones recursivas
  - Animaciones aleatorias
  - Animaciones sin fin
3. Interactividad (opcional)
  - Capturando eventos
  - Coordenadas del mouse
  - Eventos del teclado

## • | Bloque Intermedio

5. Azar y Fractales
  - Construcciones azarosas
  - Fractales
  - Fractales orgánicos

*La diferencia entre lo ordinario y lo extraordinario es ese pequeño extra*