



Proyecto 1 – Algoritmo

| | |
|--------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Resultados de Aprendizajes a Evaluar | <ul style="list-style-type: none">Analiza, mediante técnica de conteo, el orden de complejidad de un algoritmoImplementa listas enlazadas simples, dobles, circulares, filas y pilasImplementa juego simple utilizando estructura de datosRealiza experimentos manuales sobre tiempos de ejecución |
| Fecha Publicación | 26 de septiembre 2022 |
| Modalidad | Online y Grupal (3 personas) |
| Tiempo estimado de trabajo | 6-8 horas |
| Fecha de Entrega | Hasta el 23 de octubre del 2022 a las 23:59hrs. |
| Forma de Entrega | Archivo comprimido con programa Python y recursos por plataforma eFinis: Algoritmos (P) > Tareas > PROYECTO 1 Adicionalmente, durante un laboratorio y/o clase se entrevistará a los integrantes de cada grupo sobre el diseño y funcionamiento del código (se le preguntará a cualquiera de los miembros con nota grupal) |
| Instrucciones Generales | <ul style="list-style-type: none">El proyecto es grupal (3 integrantes)Puede utilizar cualquier material escrito o electrónico (no copy&paste)Es responsabilidad del/la estudiante entregar el archivo correcto o cualquier acción que deba hacer para asegurar la entrega correcta de material y respuestas solicitadas en las sección y formato en plataforma eFinis. Errores de entrega que no son solucionados antes de fecha límite reciben descuento estándar por atraso.DEBE ENTREGAR TODOS LOS ARCHIVOS EN UN SOLO ARCHIVO COMPRIMIDO .zip en sección Tarea de Algoritmo(P)Por cada día de atraso el descuento es de 1 punto (cada 24 horas desde fecha y hora del cierre de la evaluación)Tenga en cuenta los “CRITERIOS DE EVALUACIÓN” al final de este documento.La copia o la facilitación de respuestas entre compañeros es sancionada con un 1.NOTA = Suma de puntaje + 1 |

PROYECTO JUEGO SNAKE

El juego “Snake” El juego de la víbora por todos conocidos, es un juego donde inicialmente se comienza con un cuadrado que siempre está en movimiento, el cual podemos guiar hacia la derecha, izquierda, arriba o abajo. Al mismo tiempo aparecen al azar cuadrados inmóviles que cuando la víbora “se los come”, la víbora aumenta la cantidad de cuadros en 1 asemejándose a una víbora o serpiente. La víbora no puede comerse así misma ni tampoco chocar contra las paredes.

Sin embargo, ahora se requiere construir un nuevo juego Snake donde existen nuevas reglas:

- Comida Tipo 1: al comerlo, este se añade a la cola
- Comida Tipo 2: al comerlo, este se añade en la posición indicada por su número.
- Comida Tipo 3: al comerlo, se agrega a la cabeza
- Comida Tipo 4: esta comida se mueve por pantalla en forma aleatoria
- Comida Tipo 5: Cierta comida solo se puede comer si la snake encierra dicha comida haciendo tocar su cabeza con el último elemento de la cola. En ese momento la snake se invierte, todos los elementos de la lista se invierten, es decir, comienza a girar en sentido contrario y recibe bonificación de puntaje.
- Comida Tipo 6: Existen al menos, 3 pares de comida que se pueden comer en forma pareada y dan bonificación de puntaje. Entre más compleja sean enlazados los pares, mayor es la bonificación de puntaje. Por ejemplo, si estos tres pares se simbolizan por {}, [], (), entonces la comida en secuencia {{{[]()}}} * tiene mayor puntaje cuando se obtiene el ultimo par (se cierra el último paréntesis y luego se come un * que indica el final). Si la comida de este tipo no se come en secuencia correcta, entonces se realiza un descuento del puntaje.
- Usted debe decidir la forma de otorgar el puntaje, pero debe respetar el funcionamiento de comida Tipo 5 y 6.
- Usted decide la forma gráfica y sonora del juego.
- La posición y tipo de la comida se genera al azar y existe más de una comida al mismo tiempo
- Las reglas del juego pueden ser adaptadas para mejorar la jugabilidad, pero no simplificadas.

Aspectos Algorítmicos general:

- No puede utilizar ningún tipo de estructura de datos provista por el lenguaje o que sean parte del retorno de funciones/métodos (listas, vectores, matrices, diccionarios, etc.), pues usted debe implementar cualquiera que necesite mediante listas enlazadas dobles y simples.
- Toda la funcionalidad de la snake es una representación gráfica 1 a 1 de la estructura de dato que la contiene.
- No se puede utilizar ningún tipo de algoritmo de ordenamiento provisto por el lenguaje, debe implementar el que usted necesite si fuese necesario.
- Utilice las capacidades de orientación al objeto del lenguaje
- Se debe implementar, por lo menos, un tipo de lista enlazada simple y un tipo de lista enlazada doble

- Salvo el módulo sys, pygame, math, random y time, no se permite la utilización de ningún modulo o funcionalidad adicional

Recursos:

- Ejemplo de uso de pygame y un posible código base para el juego en la sección Documentos de Algoritmo (P)
- Instalar módulo pygame para Python <https://www.pygame.org/>
- Pueden obtener sonidos gratis para este tipo de juegos en <https://freesound.org/>

Restricciones: *Las restricciones lista aquellas condiciones que deben cumplirse para poder acceder a la revisión del proyecto. No se revisarán programas que no cumplan las restricciones.*

- Se prohíbe comentar el código de cualquier forma (prohibidos los comentarios o documentación)
- El programa debe ser un único archivo .py
- Programas que no puedan ejecutarse (cualquier motivo) no serán revisados

CRITERIOS DE EVALUACIÓN

Siga la siguiente pauta de evaluación para cumplir con los requerimientos. Recuerde

| Criterio | Ptje |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| Implementa y usa correctamente, al menos, una lista enlazada simple | 0.5 |
| Implementa y usa correctamente, al menos, una lista enlazada doble | 0.5 |
| La Snake esta compuesta por una estructura de datos implementada adhoc y corresponde 1 a 1 con lo que muestra el juego | 0.5 |
| El juego implementa los 4 tipos de comida básicos correctamente | 0.5 |
| El juego implementa el tipo 5 de comida que se debe encerrar y revierte a la snake | 0.5 |
| El juego implementa el tipo 6 comida pareada correctamente | 1 |
| Originalidad (el juego es bastante conocido, incluir elementos funcionales, gráficos, sonoros que sean adicionales a lo requerido que sean creaciones propias del estudiante y no copias de Internet) | 0.3 |
| Utiliza principios de O.O. correctamente | 0.2 |
| Demuestra dominio del programa y código fuente ante preguntas del profesor | 2 |
| Incumple con las restricciones algorítmicas o generales (no usar ningún tipo de lista o estructura de Python, algoritmo de ordenamiento, etc.) | -2 |
| El juego o el código es copia de otro juego o contiene porciones considerables de copia de otro compañero o solución de Internet | -2 |

Nota = Suma Puntaje +1