

Inteligencia Artificial

Andrés Gómez de Silva Garza

Proyecto de Programación #2

El proyecto se puede (y se recomienda) hacer en equipos de hasta cuatro personas. Es la responsabilidad de cada uno de los integrantes del equipo asegurarse de que la calidad de cada una de las componentes del proyecto sea suficiente para asegurar la obtención de una buena calificación. Poder completar el proyecto exitosamente requerirá de una buena organización, planeación, y coordinación entre todos los integrantes del equipo por anticipado, pues no es un proyecto trivial, así como del compromiso con el resto del equipo por parte de cada integrante. La correcta elaboración del proyecto también requerirá de la investigación independiente de algunos temas que no se hayan tratado en clase, o que se hayan tocado sólo brevemente.

La entrega consiste de tres partes:

- un programa escrito en SWI-Prolog (comentado/documentado adecuadamente para entender su funcionamiento y diseño), listo para ejecutarse,
- una presentación oral ante el resto del salón (de 30 minutos de duración), apoyándose en la proyección de una presentación de PowerPoint que incluya los puntos clave del proyecto a exponer, en la que se tiene que explicar el proyecto que se realizó (la teoría detrás de él), el diseño del programa (es decir, cómo se implementó/aplicó la teoría) y mostrar el programa en acción (todos los integrantes del equipo deben participar en la presentación oral), y
- un reporte escrito documentando la elaboración del proyecto (estructurado como un artículo científico o una mini-tesis, incluyendo conclusiones y bibliografía) y lo que se aprendió de las investigaciones independientes que se hayan tenido que hacer para realizar el proyecto.

De estas tres partes, las primeras dos se calificarán juntas y la tercera se calificará por separado. La calificación final del proyecto consistirá del promedio de las dos calificaciones, dándole el mismo peso a cada una de ellas. Las dos rúbricas que se van a utilizar para calificar las últimas dos partes estarán disponibles a través de ComunidadITAM. La rúbrica para la presentación oral se va a usar para calificar tanto la presentación oral como el programa (pues algunos de los renglones de dicha rúbrica son específicos al material visual o la forma de exponer, pero otros se pueden interpretar de manera más amplia, por ejemplo los que tienen que ver con la cantidad de detalle y con la inclusión de suficientes ejemplos en la presentación). La calificación de la presentación oral podría ser diferente para cada integrante de un equipo. En los demás aspectos se calificará de manera pareja a todos los integrantes de cada equipo. Al final de cuentas, la calificación final en el proyecto será calculada mediante la siguiente fórmula: $P = \text{Presentación} * 0.5 + \text{Escrito} * 0.5$.

Posibles proyectos (cada equipo debe escoger uno de ellos...no importa si varios equipos hacen el mismo, aunque cada equipo debe trabajar por separado, lo cual debe verse reflejado en un diseño diferente de cada versión):

1. Programar y aplicar el algoritmo de búsqueda A* para encontrar distancias geográficas mínimas (y/o tiempos de desplazamiento mínimos). Parte de la investigación independiente relacionada con este proyecto será la obtención (y captura) de datos geográficos relevantes (por ejemplo, coordenadas, distancias, tiempos de recorrido, conexiones) descriptivos de un contexto geográfico lo suficientemente complejo para ser

interesante (por ejemplo, la red del metro de la Ciudad de México o alguna otra ciudad, la red de carreteras federales entre las ciudades capitales y otras ciudades grandes mexicanas o equivalentes, la red de conexiones aéreas en México o alguna región equivalente, la red de calles de alguna zona urbana o fragmento de la misma). Otra parte de la investigación independiente tiene que ver con elegir una función heurística adecuada (lo cual puede depender de la disponibilidad de datos adecuados) para la ejecución del algoritmo.

2. Programar y aplicar el algoritmo minimax con poda alfa-beta para tener un sistema capaz de jugar (¡y jugar bien!) dominó, ya sea contra uno, dos, o tres contrincantes humanos (ya sea jugando por parejas o individualmente). Parte de la investigación independiente relacionada con este proyecto tiene que ver con la elección de una función heurística apropiada para el dominó, junto con la elaboración de reglas adecuadas para tratar de inferir, para tener el conocimiento en el que se va a tener que basar la función heurística, qué pieza(s) puede(n) tener el(los) contrincante(s), ya que, al contrario de otros juegos, el dominó se juega sin información completa.

3. Programar las estrategias de búsqueda básicas (en profundidad y en amplitud) y heurísticas ("hill-climbing", "steepest ascent hill-climbing", "best-first", "beam" y "A*") que se vieron en clase y aplicarlas a un conjunto de problemas diversos dentro de un solo dominio (por ejemplo, el grafo que usamos en clase para ejemplificarlas, aumentado con la información adicional que sea necesaria para implementar A*) para hacer un análisis comparativo. Parte de la investigación independiente relacionada con este proyecto tiene que ver con la obtención de datos estadísticos (después de implementar los métodos y capturar los datos acerca del dominio de búsqueda) que permitan hacer las comparaciones entre los métodos, así como la presentación clara de los mismos datos estadísticos y conclusiones basados en ellos, para poder entender mejor las diferencias entre los métodos analizados (buscando contestar preguntas como por ejemplo cuál método es en promedio más eficiente, cuál encuentra soluciones buenas con mayor frecuencia, cuál obtiene alguna solución—independientemente de su calidad—con mayor frecuencia, cuál produce soluciones de mejor calidad, cuál requirió más tiempo programar, cuál consiste de más líneas de código, etc.).

4. Programar algún algoritmo heurístico genérico de propagación de restricciones para la resolución de problemas de criptoaritmética. Parte de la investigación independiente relacionada con este proyecto tiene que ver con proponer, programar y probar heurísticas que permitan resolver este tipo de problemas (criptoaritmética) de forma más eficiente (garantizando que se llegue a la solución final, si la hay, pero no haciendo una búsqueda "de fuerza bruta", exhaustiva, sino dirigida).

5. Algún otro proyecto equivalente en alcance, nivel de dificultad, cantidad de investigación independiente requerida, etc. (su propuesta tiene que ser aprobada por el profesor).

Recuerde que, para que el programa se considere “una solución de inteligencia artificial”, tiene que ser lo más genérico posible (aplicable a una variedad amplia de problemas relacionados y/o fácil de adaptar para lograr aplicarlo así) y estar diseñado basándose lo más posible en un análisis cuidadoso que lleve a una teoría formal, sistemática, claramente especificada acerca del dominio de aplicación del sistema.