Gopher II

Brayan Steven Burgos Delgado

Noviembre-2018

1 Explicación general del problema:

El objetivo de este programa es salvar a la mayor cantidad de castores de ser devorados por los halcones. Teniendo n castores, m halcones en coordenadas rectangulares (x,y), sabiendo también que los castores son vulnerables por s segundos y que tienen una velocidad v, se debe encontrar la forma de salvar la mayoría de castores con un algoritmo de grafos, pero la respuesta del algoritmo debe ser los castores VULNERABLES, según sea el caso.

2 ¿Como se va aplicar MAX-FLOW para resolver el problema?:

2.1 Definición:

Un grafo G G es un par ordenado G=(V,E) G=(V,E), donde:

V es un conjunto de vértices o nodos, y E es un conjunto de aristas o arcos, que relacionan estos nodos. Normalmente V suele ser finito. Muchos resultados importantes sobre grafos no son aplicables para grafos infinitos. Se llama orden del grafo G a su número de vértices.

Existe un flujo que viaja desde un único lugar de origen hacia un único lugar de destino a través de arcos que conectan nodos intermediarios. Los arcos tienen una capacidad máxima de flujo y se trata de enviar desde la fuente al destina la mayor cantidad posible de flujo.

2.2 Ford Fulkerson:

Este método propone buscar caminos en los que se pueda aumentar el flujo hasta que se alcance el flujo máximo, la idea es encontrar una ruta de penetración con un flujo positivo neto que una los nodos de origen y destino. El flujo es siempre positivo y con unidades enteras.

El flujo a través de un arco es menor o igual que la capacidad. El flujo que entra en un nodo es igual al que sale de él.

2.3 Edmonds-Karp:

En ciencias de la computación y teoría de grafos, el Algoritmo de Edmonds-Karp es una implementación del método de Ford-Fulkerson para calcular el flujo maximal en una red de flujo con complejidad O.

2.4 Estrategia usando MAX FLOW:

2.4.1 Encontrar la coincidencia máxima entre los agujeros

2.4.2 El resultado es la resta entre el numero total de castores (n) y el numero de castores que escapan, después de aplicar MAX FLOW en totalidad

2.4.3 si s*v es mayor o igual distancia(i,j) donde i,j son las distancias entre un castor y un agujero

3 Representación gráfica explicando la respuesta de un caso de prueba:

3.1 caso de entrada que se usara:

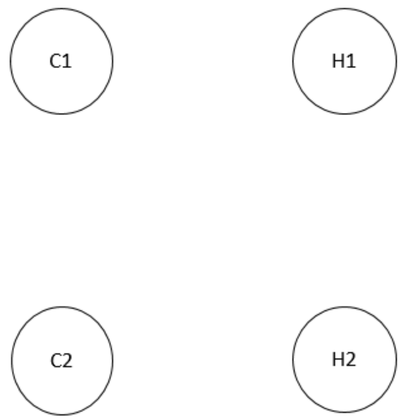
2 2 13 6
16.0 8.0
506.0 236.0
127.0 82.0
8.0 13.0
Caso de prueba

sabiendo ya el caso de prueba se tiene que S*V=78

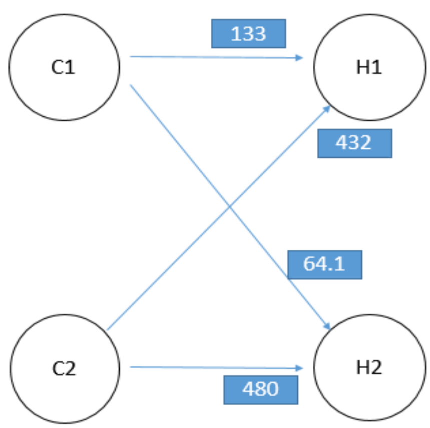
Termino	Castores(x,y)	Agujeros(x,y)	Distancia(m)
c1,h1	16,8	127,82	133
c2,h2	16,8	80,13	64.1
c2,h1	506,236	127,82	432
c2,h2	506,236	80,13	480

Table 1: tabla de distancias desde los agujeros a los castores.

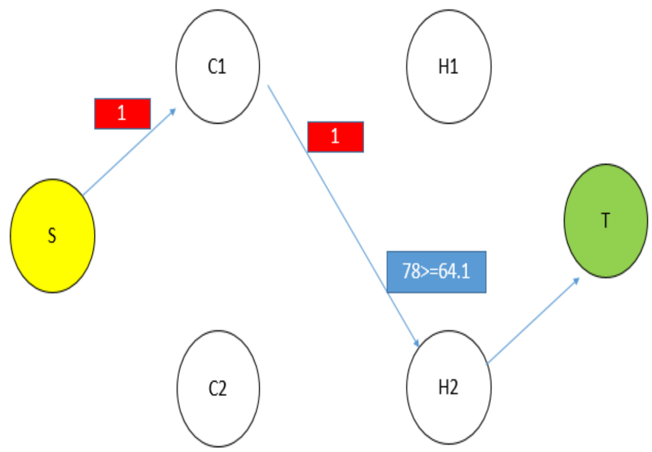
Creacion de los nodos ya establecidos en la Tabla 1:



Creacion de las distancias y lo ”arcos” que son simplemente para representar la distancia:



Luego usamos el nodo de inicio y el nodo final para que el algoritmo de MAX FLOW nos arroje el resultado final y así obtener un numero x y la respuesta es n-x y en este caso simplemente arroja el flujo de 1 que el único que se salva y por ello el numero de los que son vulnerables es solo UNO (1).



Referencias

1.http://flujomaximo.blogspot.com/
2.https://www.udebug.com/UVa/10080/
3.http://www.algorithmist.com/index.php/$\text{UVa}_1$0080/
4.$https : //www.youtube.com/watch?v = DRwWwwyl3CI/$
5.$https : //www.youtube.com/watch?v = T4jlNZjCaWk/$