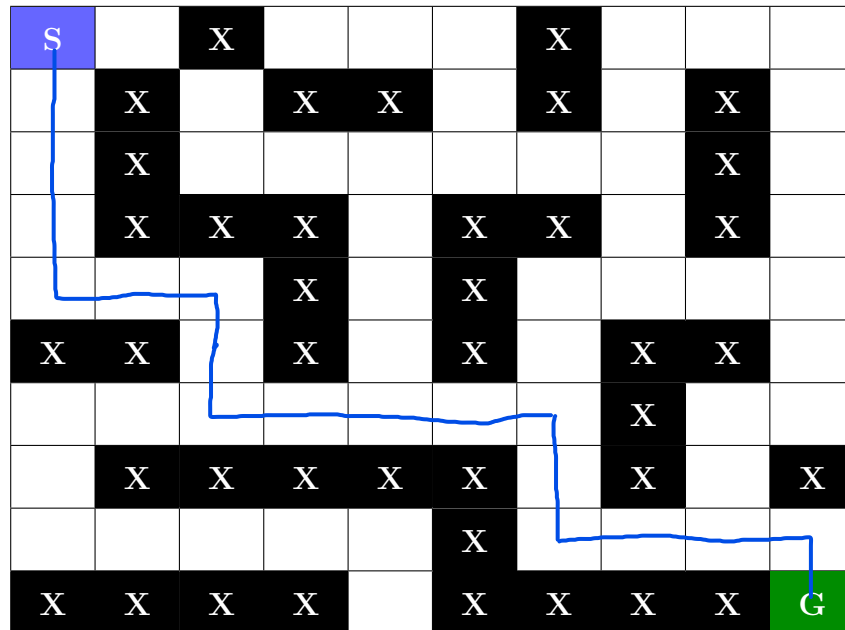


Actividad 2 – Búsqueda no informada (BFS vs DFS)

Laberinto 10×10 – Estado inicial S , objetivo G , paredes X . Costo uniforme = 1.

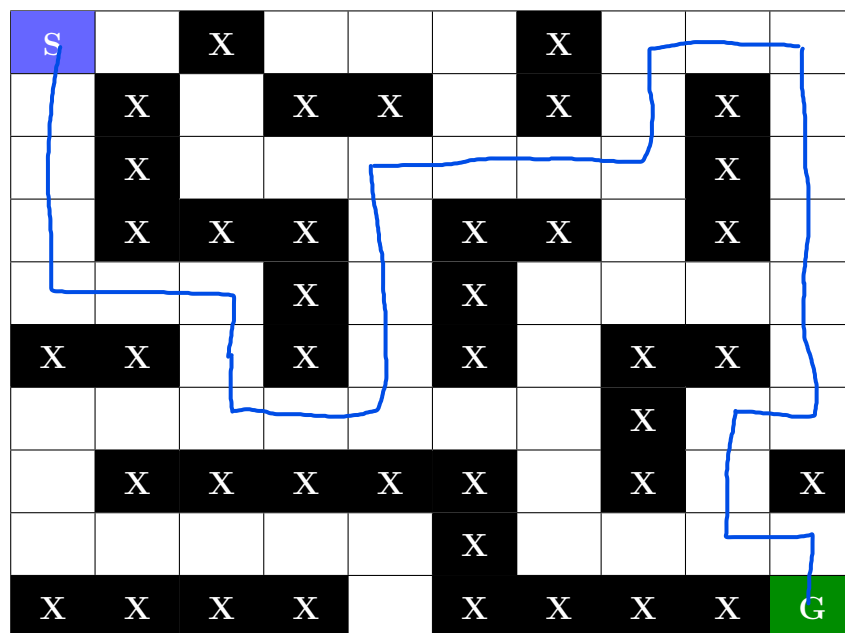
A. Marque orden de expansión y camino (BFS)



Indique: orden de expansión (1,2,3,...) y dibuje el *camino final*.

Costo total del camino (número de pasos) = 18.

B. Marque orden de expansión y camino (DFS)



Indique: orden de expansión (1,2,3,...) y dibuje el *camino final*.

Costo total del camino (número de pasos) = 32.

C. Comparación

| | BFS | DFS |
|----------------------------------|-------------|-----------------------------------|
| Estructura de datos principal | Cola (FIFO) | Pila (LIFO) |
| <i>Completo</i> (espacio finito) | Sí | Con control de ciclos/profundidad |
| <i>Óptimo</i> (costos = 1) | Sí | No necesariamente |
| Uso de memoria | Alto | Bajo |
| Camino hallado (nº de pasos) | <u>18</u> | <u>32</u> |

Sugerencia: Numere el orden de expansión dentro de cada celda y trace el camino final con una línea clara.

En este ejercicio se compararon los algoritmos de búsqueda BFS y DFS aplicados a un laberinto con costo uniforme. Se observó que BFS siempre encuentra el camino más corto hacia la meta, aunque utiliza más memoria, mientras que DFS puede encontrar caminos mucho más largos y no garantiza la solución óptima, pero requiere menos memoria. Por lo tanto, para problemas donde es importante la optimalidad y el costo de los pasos es uniforme, BFS es la mejor opción. Sin embargo, si la memoria es limitada y no es necesario el camino más corto, DFS puede ser útil. Esta comparación evidencia la importancia de elegir el algoritmo adecuado según las características y restricciones del problema.