

1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and J is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

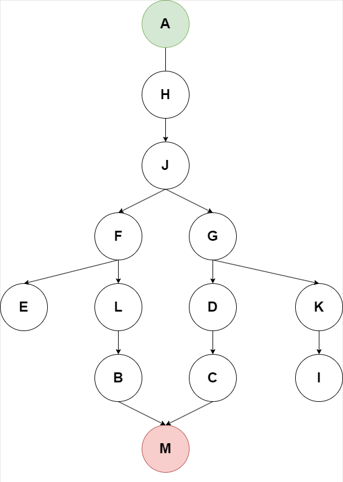
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and M is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

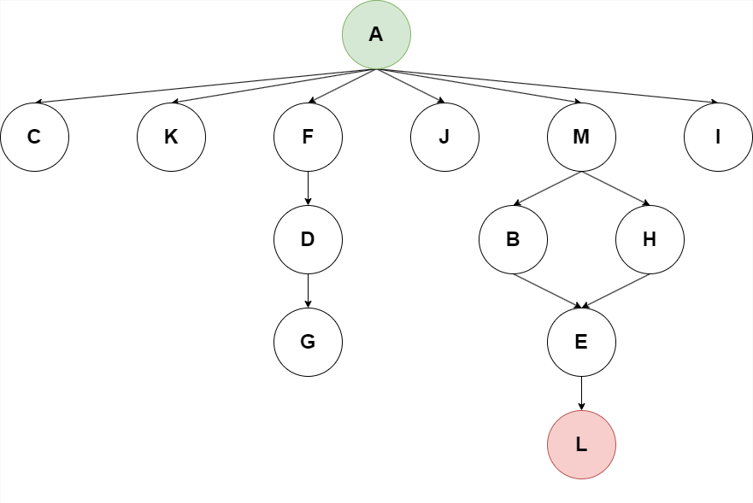
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and L is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

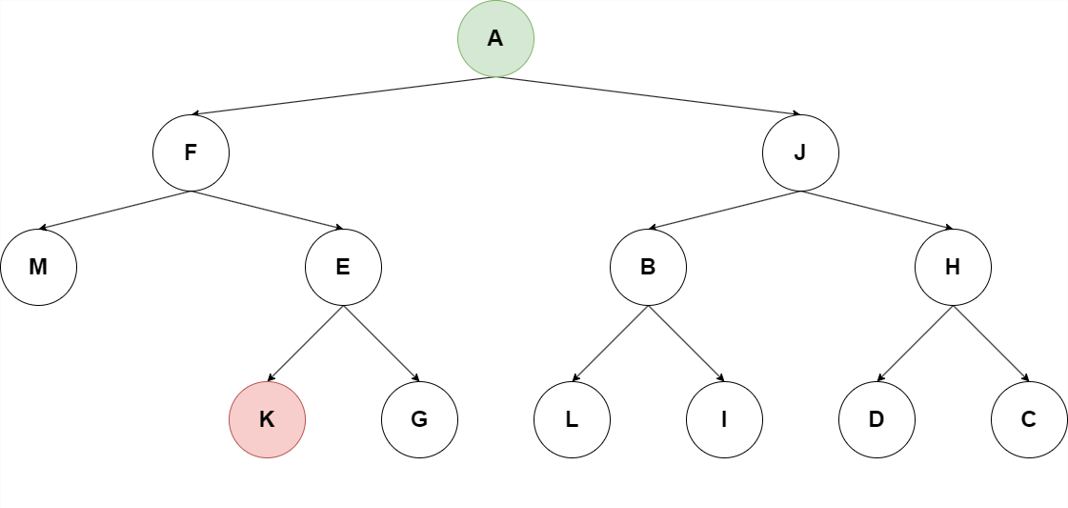
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and K is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

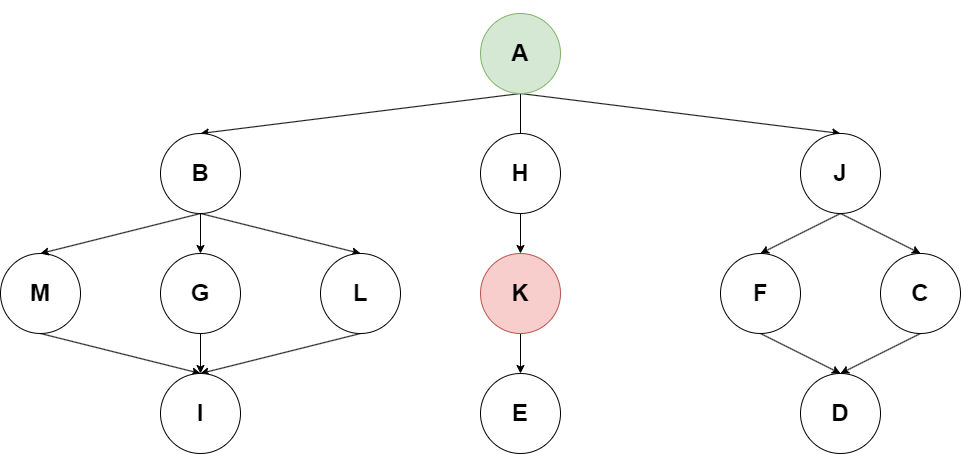
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and K is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

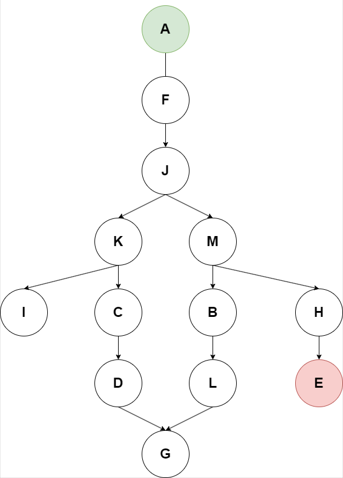
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and E is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

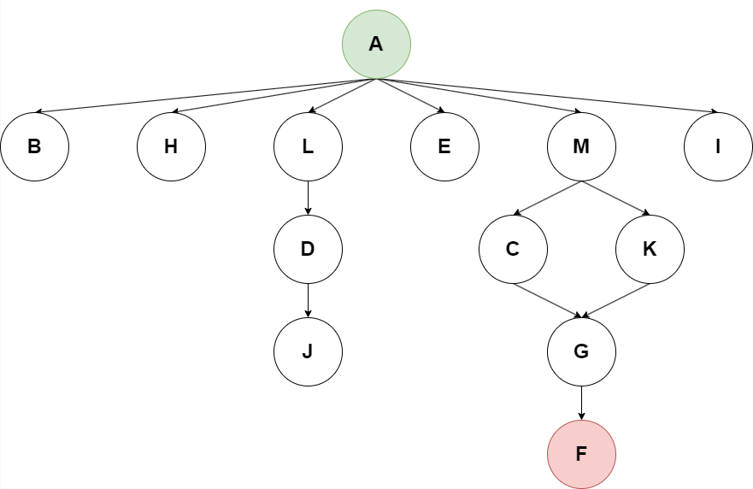
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and F is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

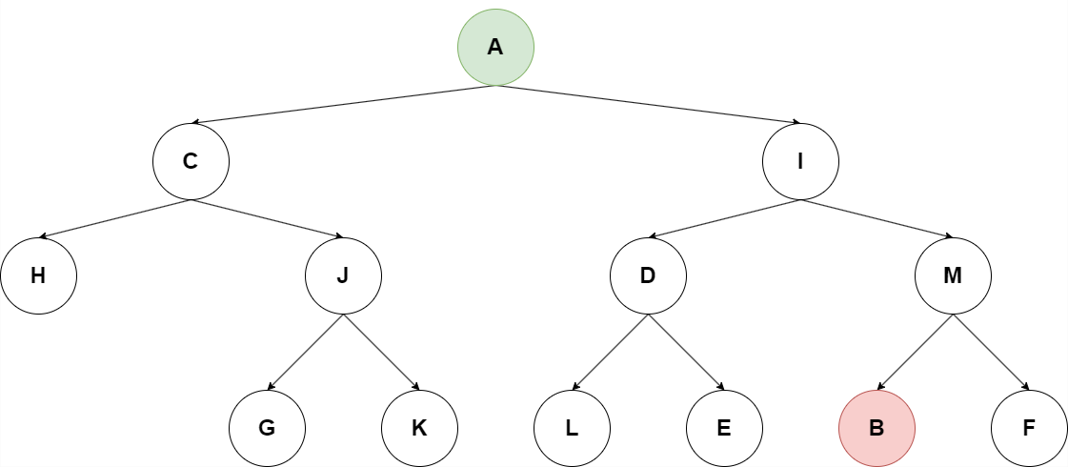
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and B is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

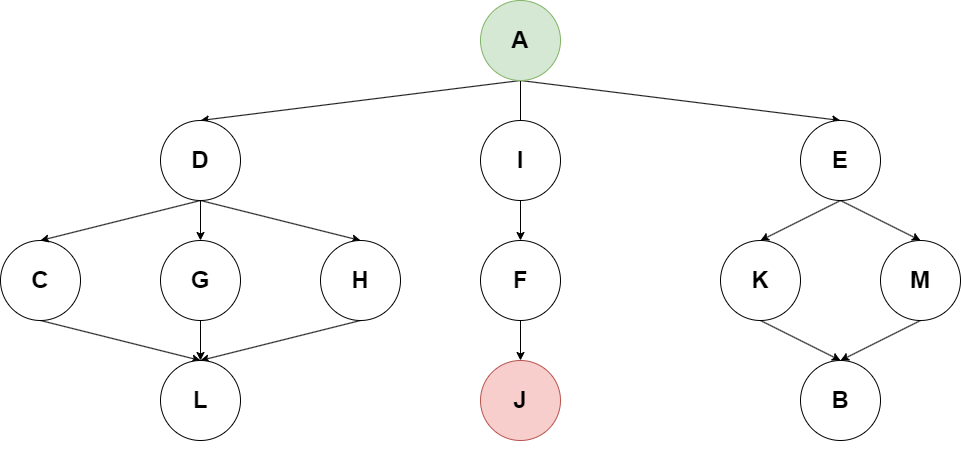
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and J is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

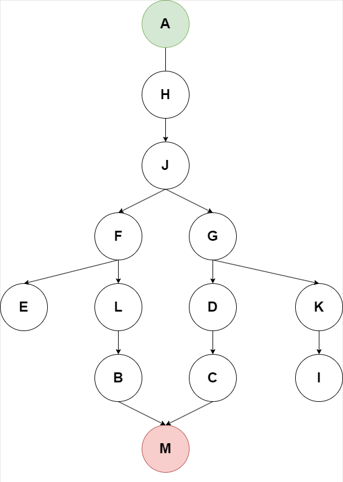
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and M is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

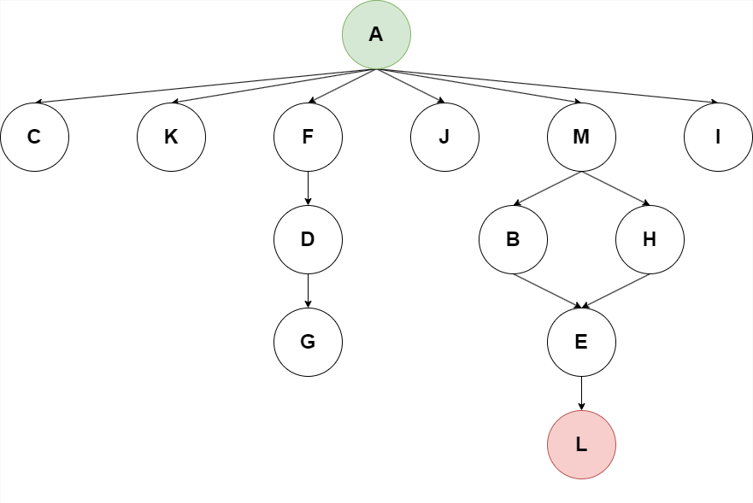
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and L is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

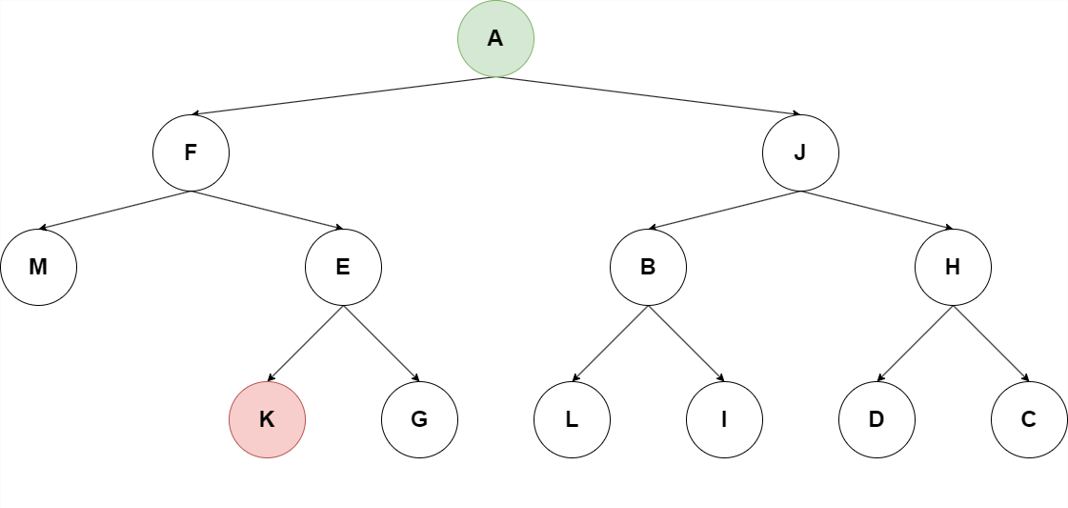
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and K is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

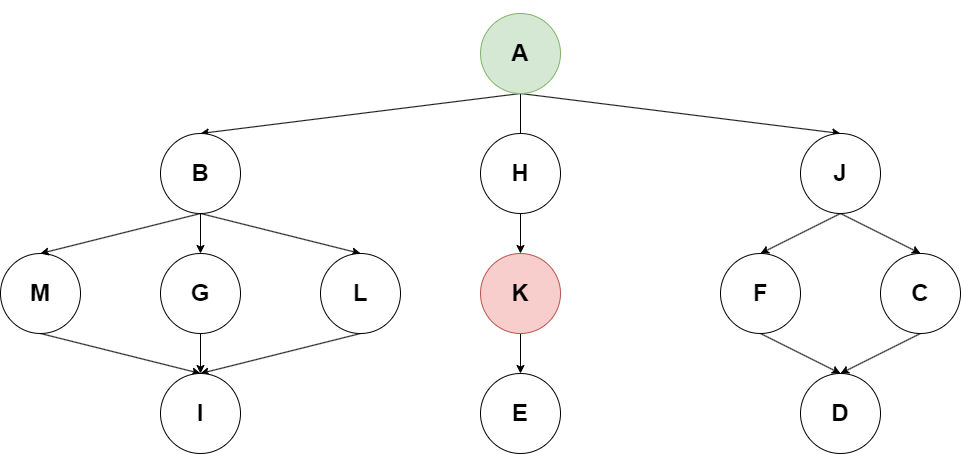
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and K is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

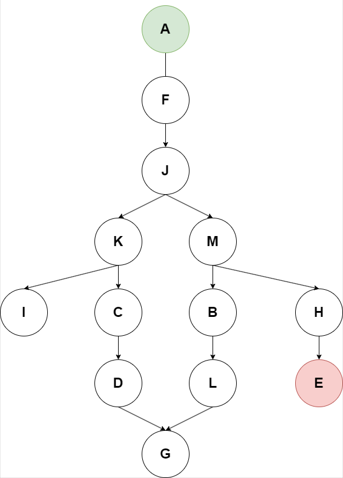
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and E is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

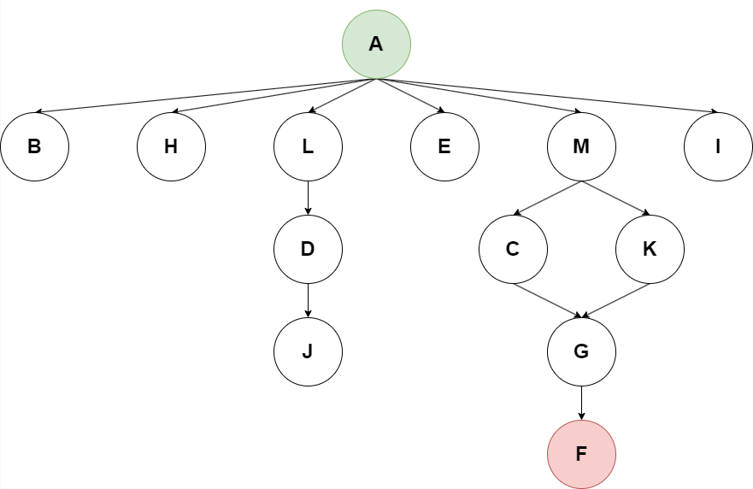
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and F is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

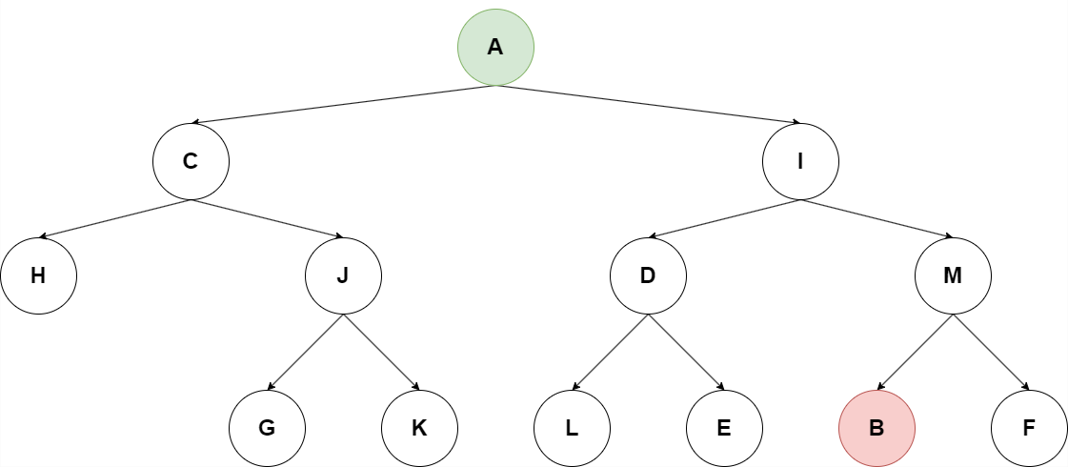
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and B is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

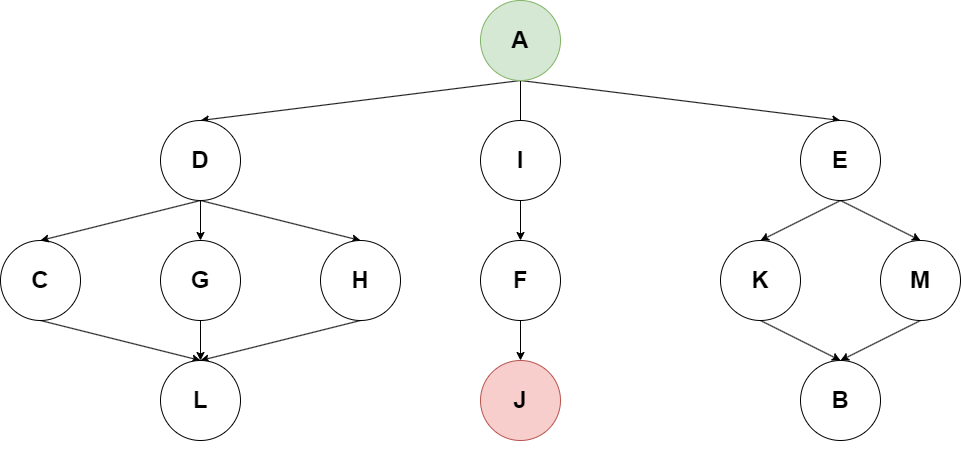
|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**



1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and J is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**

A képen kör látható

Automatikusan generált leírás

1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and M is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**

A képen diagram látható

Automatikusan generált leírás

1. (1.5 points) Write a possible result path of the Breadth-first graph search algorithm for the given tree, where A is the starting node and L is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**

A képen diagram látható

Automatikusan generált leírás

1. (1.5 points) Write a possible result path of the Depth-first graph search algorithm for the given tree, where A is the starting node and K is the goal node.

2. (8.5 points) Given a 6x6 chess board, walk through each cell with a knight starting from the top-left corner. Rules are that you can only touch a field once (you can’t move on a field where you were before) and can only move in L-shape.

Allowed movements:

A képen keresztrejtvény, csempézett látható

Automatikusan generált leírás

1. **Characteristics:**

|  |  |
| --- | --- |
|  | 0 – the knight was not on this field yet  1 – the current position of the knight  2 – the knight was on this field already |

1. **Set of valid states:**
2. **Initial state:**
3. **Set of goal states:**
4. **Set of operators:**

– is the row index of the knight; – is the column index of the knight;

– is the new row index of the knight; – is the new column index of the knight;

1. **Domain of operators:**
2. **Effect definition (transition function):**