CS5201: Advanced Artificial Intelligence

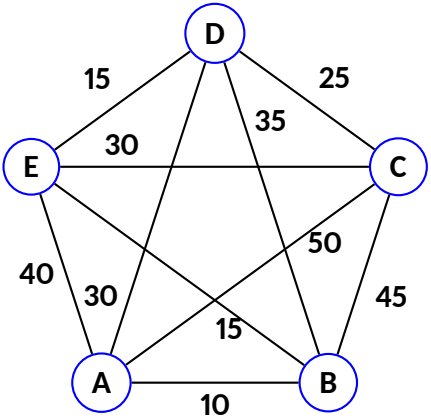
Heuristic search

Arijit Mondal
Dept of Computer Science and Engineering
Indian Institute of Technology Patna
www.iitp.ac.in/~arijit/

State space modeling

- State or configurations
  - A set of variables which define a state or configuration
  - Domains for every variable and constraints among variables to define a valid configuration
- State transformation rules or moves
  - A set of rules which define which are the valid set of next state of a given state
  - It also indicates who can make these moves (OR Nodes, AND nodes, etc)
- State space or implicit graph
  - The Complete Graph produced out of the state transformation rules.
  - Typically too large to store. Could be Infinite.
- Start and goal states
- Solutions, costs
  - Depending on the problem formulation, it can be a PATH from Start to Goal or a Sub-graph of And-ed Nodes
- Search algorithms
  - Intelligently explore the Implicit Graph or State Space by examining only a small sub-set to find the solution
  - To use Domain Knowledge or HEURISTICS to try and reach Goals faster

Example problems



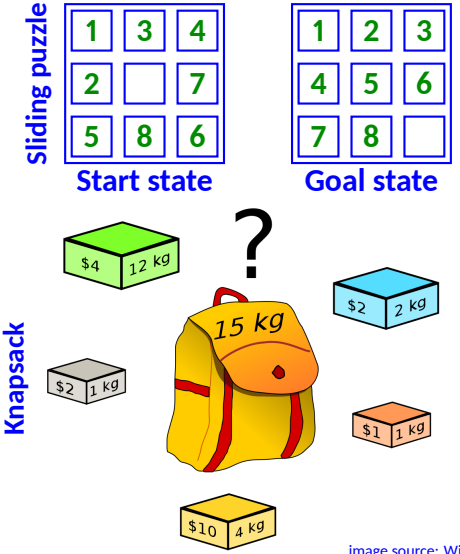
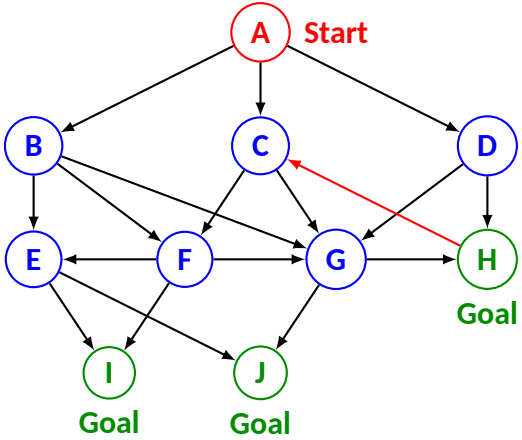


image source: Wikipedia

Searching implicit graph: Algorithms

- **Basic algorithms — Depth-First (DFS), Breadth-First (BFS), Iterative deepening (IDS)**
- **Cost-based algorithms — Depth-First Branch-and-Bound, Best First Search, Best-First Iterative deepening**
- **Widely used algorithms — A* and IDA* (OR graphs), AO* (AND/OR graphs), Alpha-beta pruning (Game-trees)**

Searching state space graph



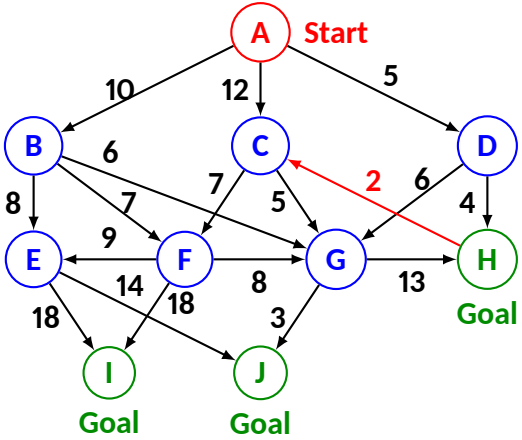
- Depth-first search
- Breadth-first search
- Iterative deepening search

Basic algorithms

1. [Initialize] Initially the OPEN List contains the Start Node s. CLOSED List is Empty.
2. [Select] Select the first Node n on the OPEN List. If OPEN is empty, Terminate
3. [Goal Test] If n is Goal, then decide on Termination or Continuation / Cost Updation
4. [Expand]
   a. Generate the successors $n_1, n_2, \ldots, n_k$, of node n, based on the State Transformation Rules
   b. Put n in LIST CLOSED
   c. For each n_i, not already in OPEN or CLOSED List, put n_i in the FRONT (for DFS) / END (for BFS) of OPEN List
   d. For each n_i already in OPEN or CLOSED decide based on cost of the paths
5. [Continue] Go to Step 2

Algorithm IDS Performs DFS level by level iteratively (DFS(1), DFS(2), ...and so on)

Searching state space graph with edge cost



Cost ordered search:

- DFBB
- Best first search
- Best first IDS
- Use of heuristic estimates: A*, AO*

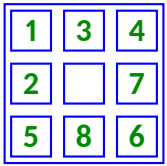
Heuristic search

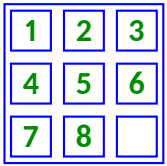
- State or configurations
- State transformation rules or moves
- State space or implicit graph
- Start and goal states
- Solutions, costs

Heuristic search

- State or configurations
- State transformation rules or moves
- State space or implicit graph
- Start and goal states
- Solutions, costs
- Heuristics
  - Estimates of cost from a given state to goal. This, along with the current cost of the path from start till now is used to guide the search
- Heuristic search algorithms
  - Algorithm A*, Depth-First Branch & Bound, IDA*, AO*, Alpha-Beta, etc.
- $g(n)$ – current cost of the node from start
- $h(n)$ – estimated cost of the best path from n to goal
- $f(n) = g(n) + h(n)$ – heuristic estimate

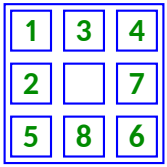
Example of heuristics: 8 puzzle problem



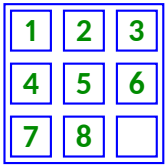


Example of heuristics: 8 puzzle problem

- **Heuristic 1: Number of misplaced tiles**



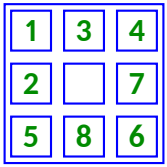
Start state



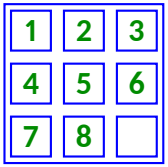
Goal state

Example of heuristics: 8 puzzle problem

- Heuristic 1: Number of misplaced tiles
- Heuristic 2: Sum of tile distance from goal

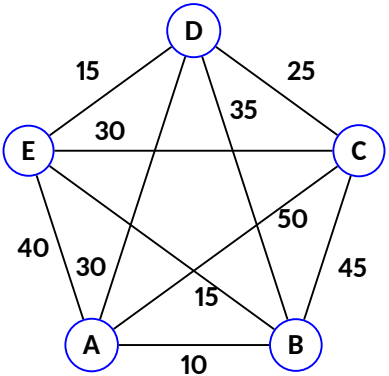


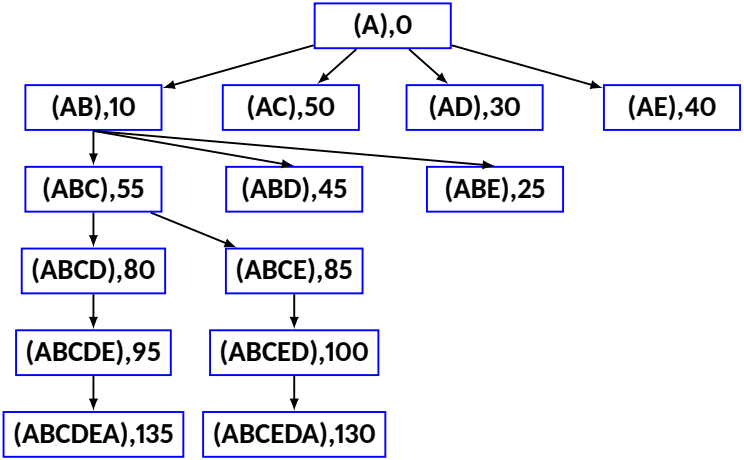
Start state



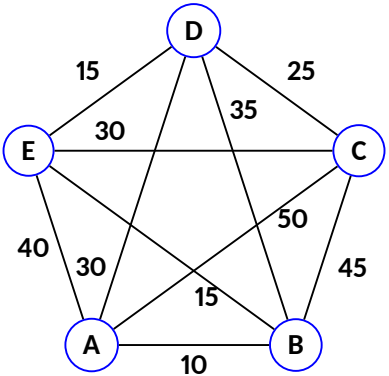
Goal state

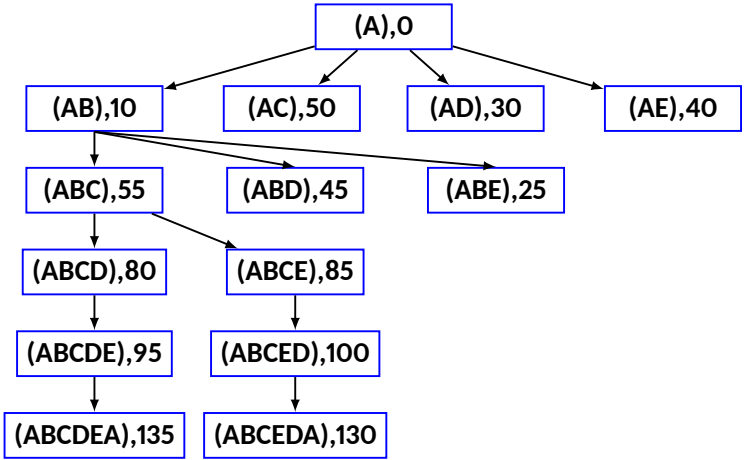
OR Graph: Travelling salesperson problem





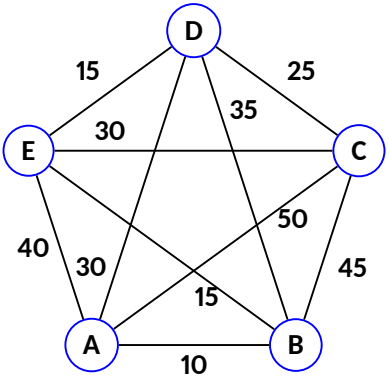
OR Graph: Travelling salesperson problem

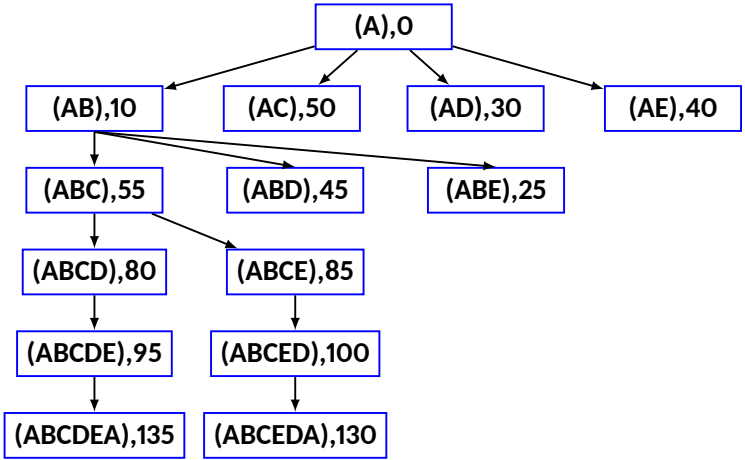




- **Heuristic 1: Shortest path**

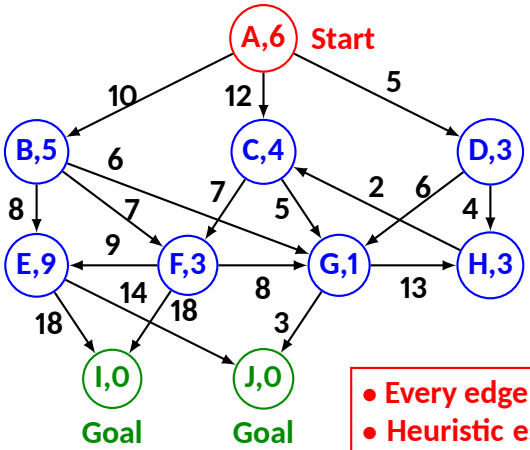
OR Graph: Travelling salesperson problem





- Heuristic 1: Shortest path
- Heuristic 2: Minimum spanning tree

Searching implicit graph: Edge & Heuristic costs



Cost ordered search:

- **DFBB**
- **A* - Best first search**
- **Best first IDS - IDA***
- **Use of heuristic estimates: A*, AO***

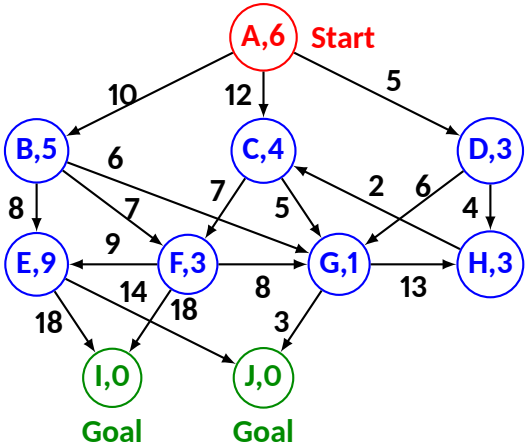
- **Every edge (n, m) has a cost $c(n, m) > 0$**
- **Heuristic estimate: $h(n) \geq 0$ at every node is the estimated cost of the minimum cost path from node n to goal**

Algorithm A*

- **Each Node n in the algorithm has a cost $g(n)$ and a heuristic estimate $h(n)$, $f(n) = g(n) + h(n)$. Assume all $c(n, m) > 0$**

1. **[Initialize] Initially the OPEN List contains the Start Node s. $g(s) = 0$, $f(s) = h(s)$. CLOSED List is Empty.**
2. **[Select] Select the Node n on the OPEN List with minimum $f(n)$. If OPEN is empty, Terminate with Failure**
3. **[Goal Test, Terminate] If n is Goal, then Terminate with Success and path from s to n.**
4. **[Expand]**
   a. **Generate the successors $n_1, n_2, \ldots, n_k$, of node n, based on the State Transformation Rules**
   b. **Put n in CLOSED List**
   c. **For each n_i, not already in OPEN or CLOSED List, compute $g(n_i) = g(n) + c(n, n_i), f(n_i) = g(n_i) + h(n_i)$, Put n_i in the OPEN List**
   d. **For each n_i already in OPEN, if $g(n_i) > g(n) + c(n, n_i)$, then revise costs as: $g(n_i) = g(n) + c(n, n_i), f(n_i) = g(n_i) + h(n_i)$**
5. **[Continue] Go to Step 2**

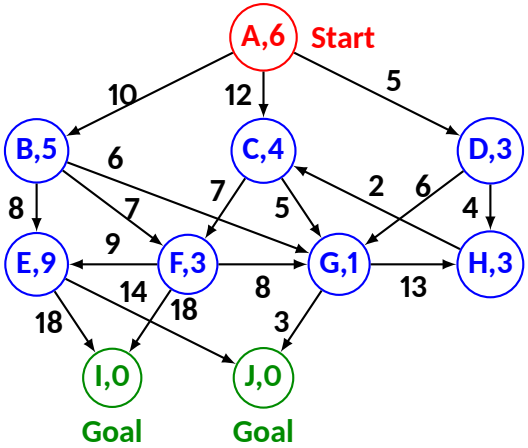
Execution of A*



(node, $g(n), h(n), f(n)$), O–OPEN, C–CLOSED

St.	Outcome

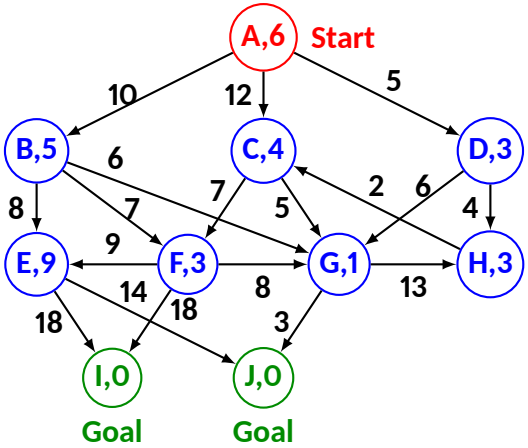
Execution of A*



(node, $g(n), h(n), f(n)$), O–OPEN, C–CLOSED

St.	Outcome
1	O={(A,0,6,6)}, C={}

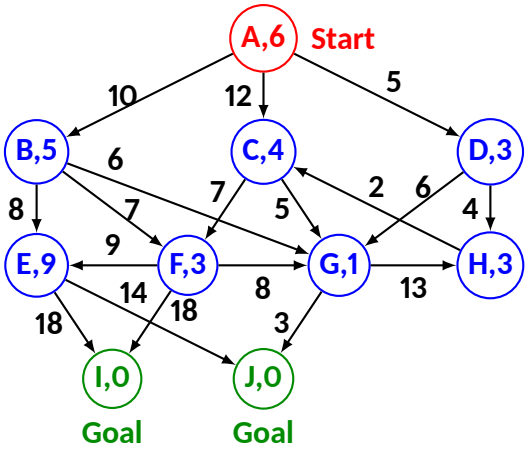
Execution of A*



(node, $g(n), h(n), f(n)$), O–OPEN, C–CLOSED

St.	Outcome
1	O={(A,0,6,6)}, C={}
2	O={(B,10,5,15), (C,12,4,16), (D,5,3,8)}, C={A}

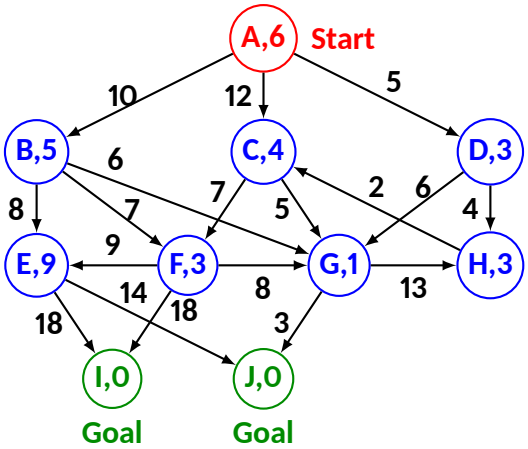
Execution of A*



(node, $g(n), h(n), f(n)$), O–OPEN, C–CLOSED

St.	Outcome
1	O={(A,0,6,6)}, C={}
2	O={(B,10,5,15), (C,12,4,16), (D,5,3,8)}, C={A}
3	O={(B,10,5,15), (C,12,4,16), (G,11,1,12), (H,9,3,12)}, C={A,D[A]}

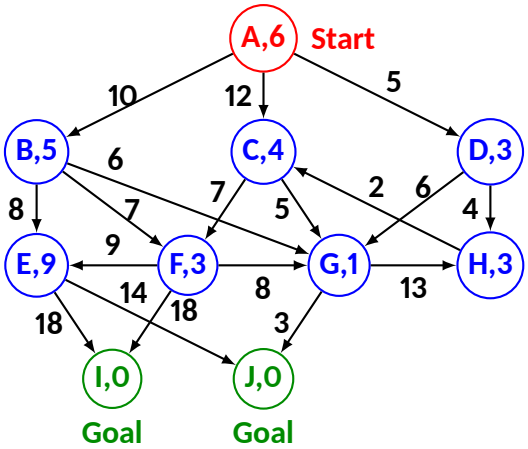
Execution of A*



(node, $g(n), h(n), f(n)$), O–OPEN, C–CLOSED

St.	Outcome
1	O={(A,0,6,6)}, C={}
2	O={(B,10,5,15), (C,12,4,16), (D,5,3,8)}, C={A}
3	O={(B,10,5,15), (C,12,4,16), (G,11,1,12), (H,9,3,12)}, C={A,D[A]}
4	O={(B,10,5,15), (C,11,4,15), (G,11,1,12)}, C={A,D[A], H[D]}

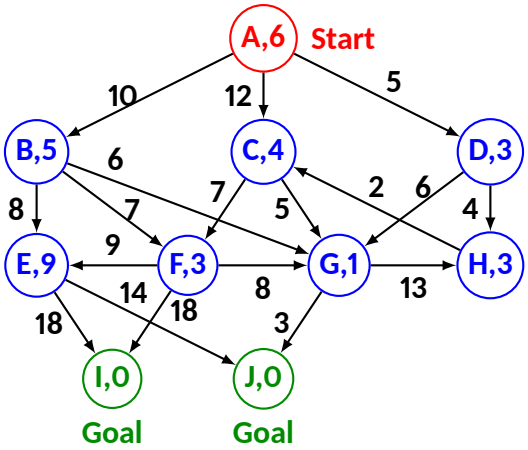
Execution of A*



(node, $g(n), h(n), f(n)$), O–OPEN, C–CLOSED

St.	Outcome
1	O={(A,0,6,6)}, C={}
2	O={(B,10,5,15), (C,12,4,16), (D,5,3,8)}, C={A}
3	O={(B,10,5,15), (C,12,4,16), (G,11,1,12), (H,9,3,12)}, C={A,D[A]}
4	O={(B,10,5,15), (C,11,4,15), (G,11,1,12)}, C={A,D[A], H[D]}
5	O={(B,10,5,15), (C,11,4,15), (J,14,0,14)}, C={A,D[A], H[D], G[D]}

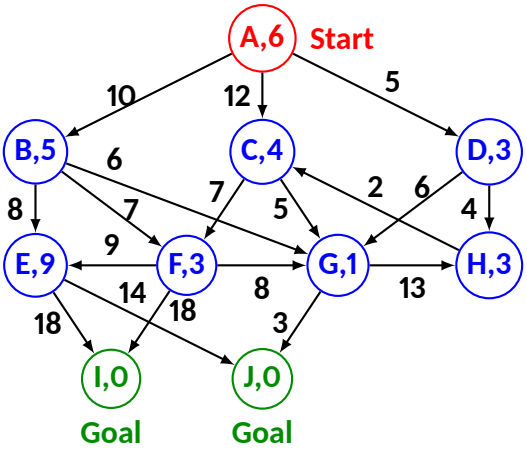
Execution of A*



(node, $g(n), h(n), f(n)$), O–OPEN, C–CLOSED

St.	Outcome
1	O={(A,0,6,6)}, C={}
2	O={(B,10,5,15), (C,12,4,16), (D,5,3,8)}, C={A}
3	O={(B,10,5,15), (C,12,4,16), (G,11,1,12), (H,9,3,12)}, C={A,D[A]}
4	O={(B,10,5,15), (C,11,4,15), (G,11,1,12)}, C={A,D[A], H[D]}
5	O={(B,10,5,15), (C,11,4,15), (J,14,0,14)}, C={A,D[A], H[D], G[D]}
6	Goal J found. Terminate with cost 14, path – A, D, G, J

Properties of A*



If lower heuristic bounds estimates and edge are costs non-negative are positive:

- First solution is optimal
- No node in closed in ever reopened
- Whenever a node is removed from open its minimum cost from start is found
- Every node n with $f(n)$ less than optimal cost is expanded
- If heuristics are more accurate then search is less

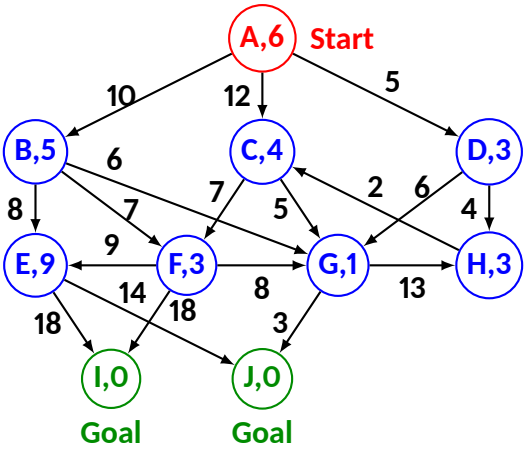
Heuristic properties

- **An admissible heuristic is one that never overestimates the cost to reach the goal**
  - **Since $g(n)$ is the actual cost to reach n, therefore $f(n) = g(n) + h(n)$ never overestimates the true cost of a goal**
- **A heuristic $h(n)$ is consistent if, for every node n and every successor n' of n generated by any action a, the estimated cost of reaching the goal from n is no greater than the step cost of getting to n' plus the estimated cost of reaching the goal from n' :** $h(n) \leq c(n, a, n') + h(n')$

Depth First Branch-and-Bound algorithm

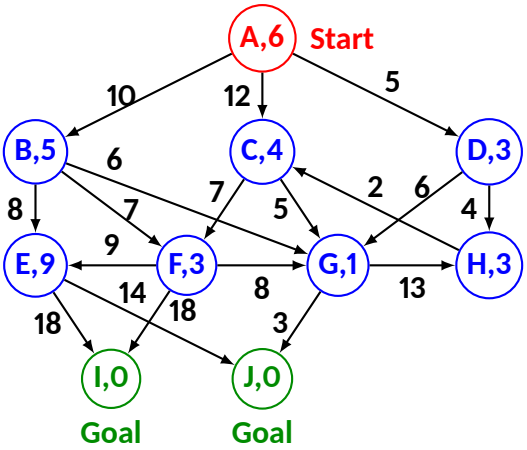
1. **Initialize Best-Cost to INFINITY**
2. **Perform DFS with costs and Backtrack from any node n whose $f(n) \geq$ Best-Cost**
3. **On reaching a Goal Node, update Best-Cost to the current best**
4. **Continue till OPEN becomes empty**

Execution of DFBB



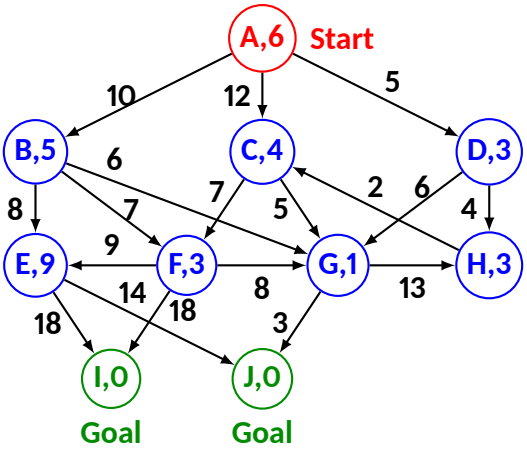
Step	OPEN	CLOSED

Execution of DFBB



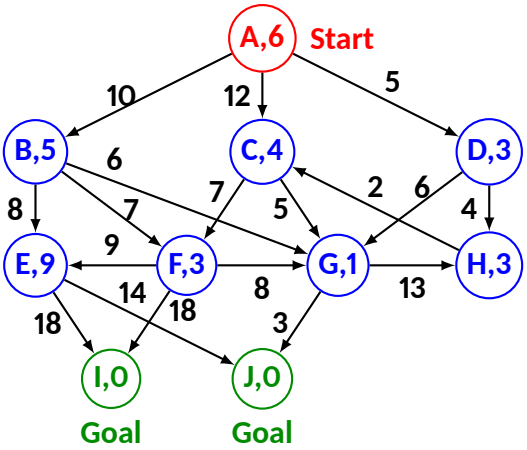
Step	OPEN	CLOSED
1	A	{}

Execution of DFBB



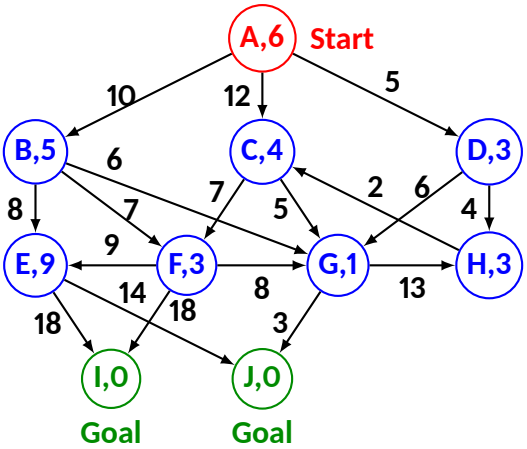
Step	OPEN	CLOSED
1	A	{}
2	(B,15), (C,16), (D,8), CB=∞	A

Execution of DFBB



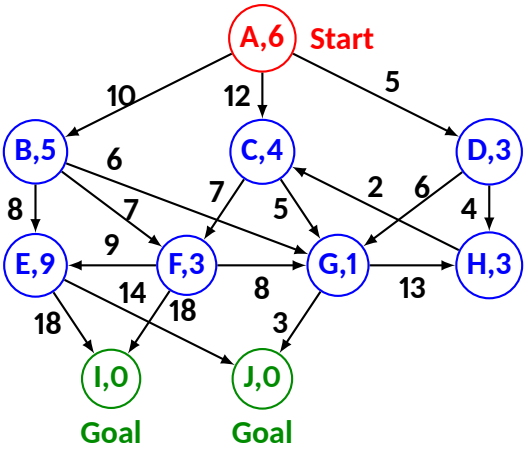
Step	OPEN	CLOSED
1	A	{}
2	(B,15), (C,16), (D,8), CB=∞	A
3	(E,27), (F,20), (G,17), (C,16), (D,8), CB=∞	B

Execution of DFBB



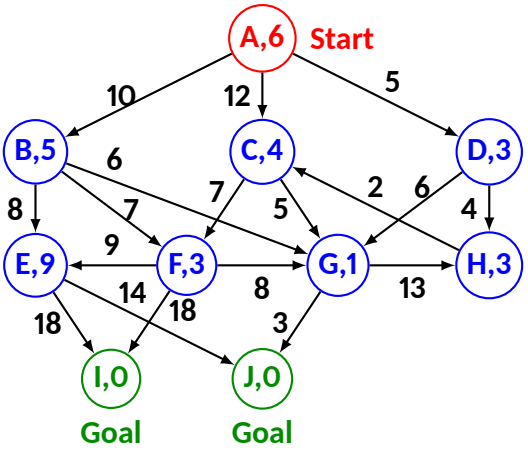
Step	OPEN	CLOSED
1	A	{}
2	(B,15), (C,16), (D,8), CB=∞	A
3	(E,27), (F,20), (G,17), (C,16), (D,8), CB=∞	B
4	(I,36), (J,32), (F,20), (G,17), (C,16), (D,8), CB=∞	E

Execution of DFBB



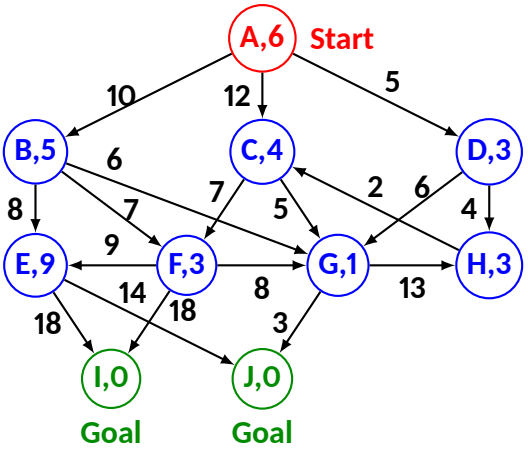
Step	OPEN	CLOSED
1	A	{}
2	(B,15), (C,16), (D,8), CB=∞	A
3	(E,27), (F,20), (G,17), (C,16), (D,8), CB=∞	B
4	(I,36), (J,32), (F,20), (G,17), (C,16), (D,8), CB=∞	E
5	(J,32), (F,20), (G,17), (C,16), (D,8), CB=36	I

Execution of DFBB



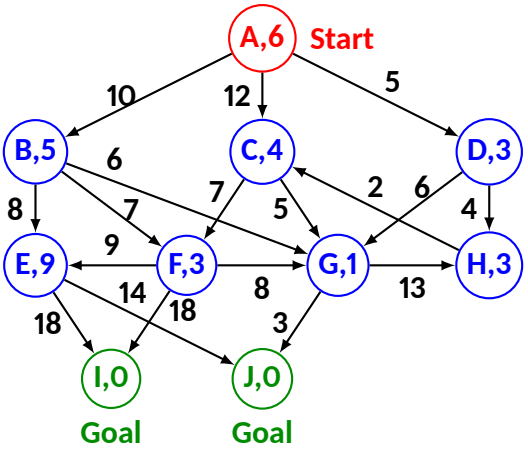
Step	OPEN	CLOSED
1	A	{}
2	(B,15), (C,16), (D,8), CB=∞	A
3	(E,27), (F,20), (G,17), (C,16), (D,8), CB=∞	B
4	(I,36), (J,32), (F,20), (G,17), (C,16), (D,8), CB=∞	E
5	(J,32), (F,20), (G,17), (C,16), (D,8), CB=36	I
6	(F,20), (G,17), (C,16), (D,8), CB=32	J

Execution of DFBB



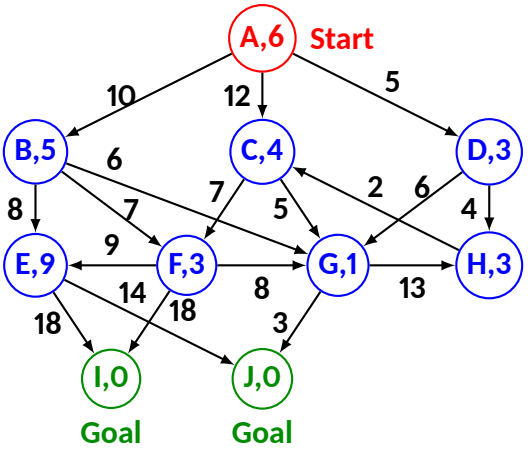
Step	OPEN	CLOSED
1	A	{}
2	(B,15), (C,16), (D,8), CB=∞	A
3	(E,27), (F,20), (G,17), (C,16), (D,8), CB=∞	B
4	(I,36), (J,32), (F,20), (G,17), (C,16), (D,8), CB=∞	E
5	(J,32), (F,20), (G,17), (C,16), (D,8), CB=36	I
6	(F,20), (G,17), (C,16), (D,8), CB=32	J
7	(G,17), (C,16), (D,8), CB=32	F

Execution of DFBB



Step	OPEN	CLOSED
1	A	{}
2	(B,15), (C,16), (D,8), CB=∞	A
3	(E,27), (F,20), (G,17), (C,16), (D,8), CB=∞	B
4	(I,36), (J,32), (F,20), (G,17), (C,16), (D,8), CB=∞	E
5	(J,32), (F,20), (G,17), (C,16), (D,8), CB=36	I
6	(F,20), (G,17), (C,16), (D,8), CB=32	J
7	(G,17), (C,16), (D,8), CB=32	F
8	(J,20), (H,33), (C,16), (D,8), CB=32	G

Execution of DFBB



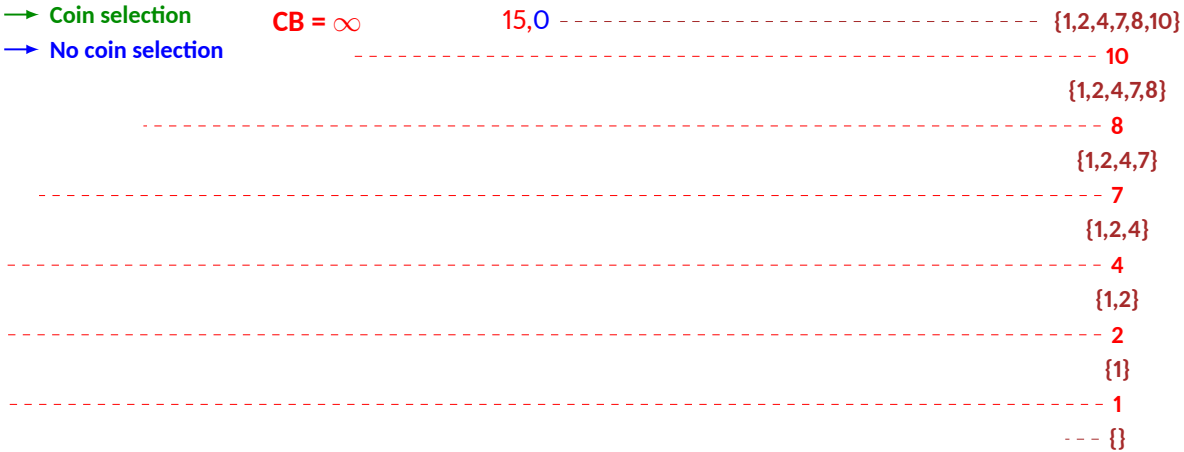
Step	OPEN	CLOSED
1	A	{}
2	(B,15), (C,16), (D,8), CB=∞	A
3	(E,27), (F,20), (G,17), (C,16), (D,8), CB=∞	B
4	(I,36), (J,32), (F,20), (G,17), (C,16), (D,8), CB=∞	E
5	(J,32), (F,20), (G,17), (C,16), (D,8), CB=36	I
6	(F,20), (G,17), (C,16), (D,8), CB=32	J
7	(G,17), (C,16), (D,8), CB=32	F
8	(J,20), (H,33), (C,16), (D,8), CB=32	G
...	..., CB=20	J

Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$

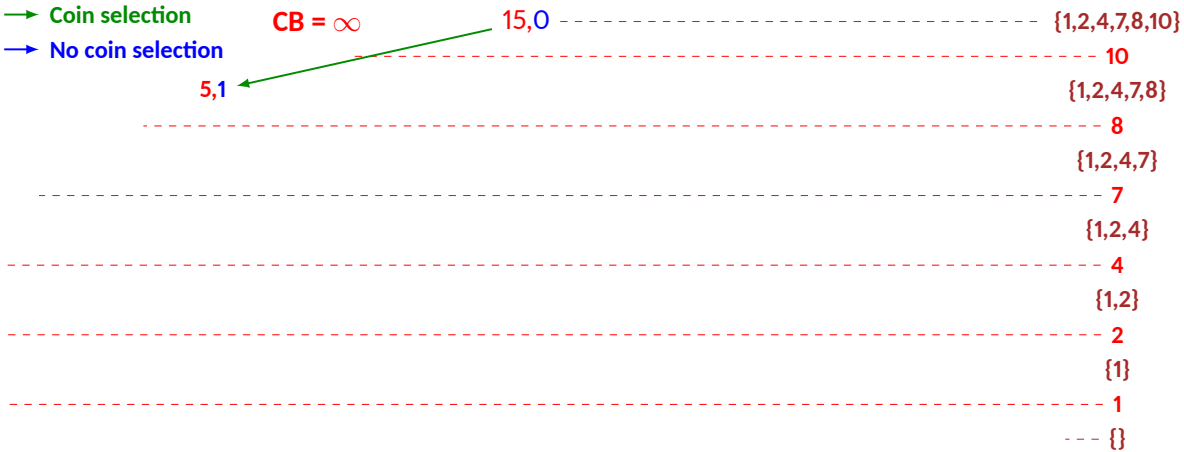
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



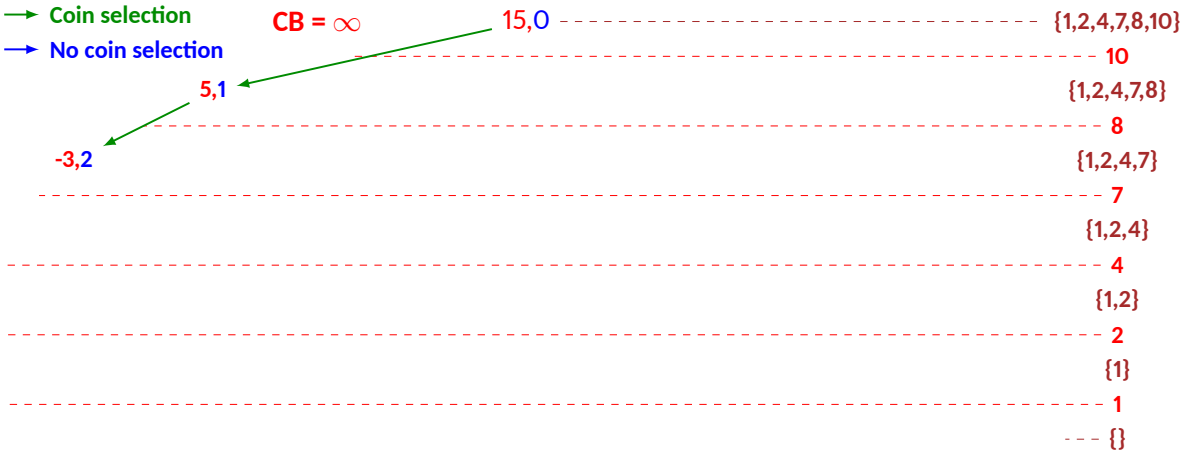
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



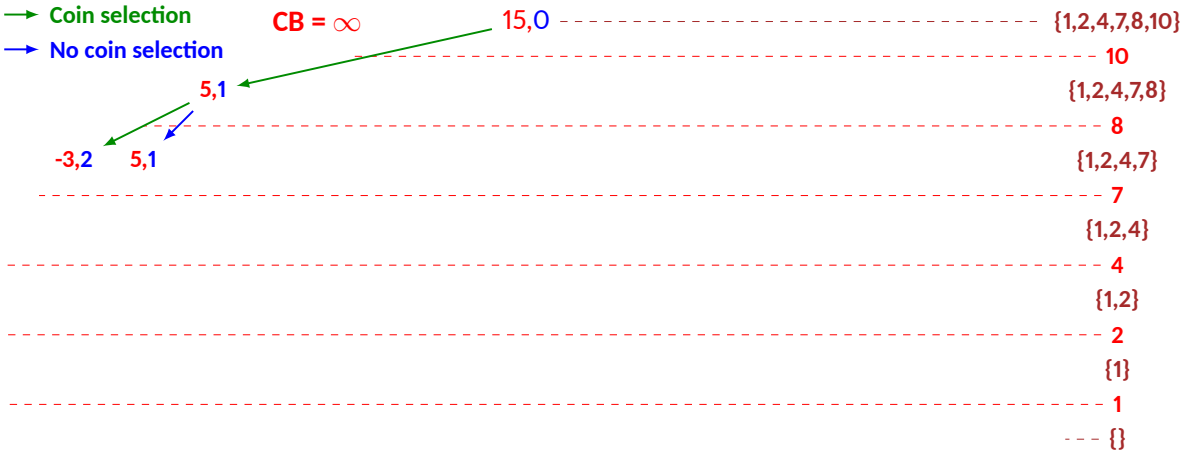
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



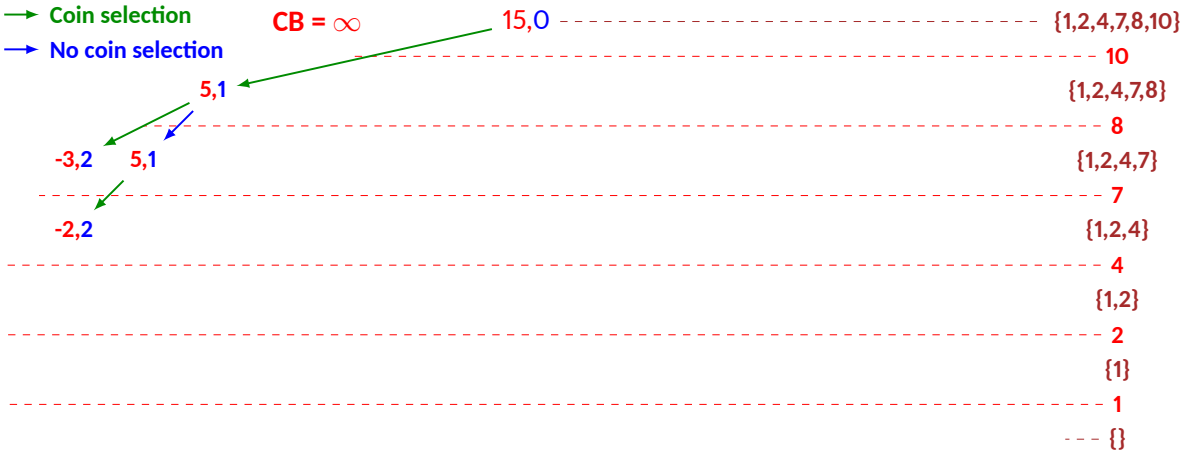
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



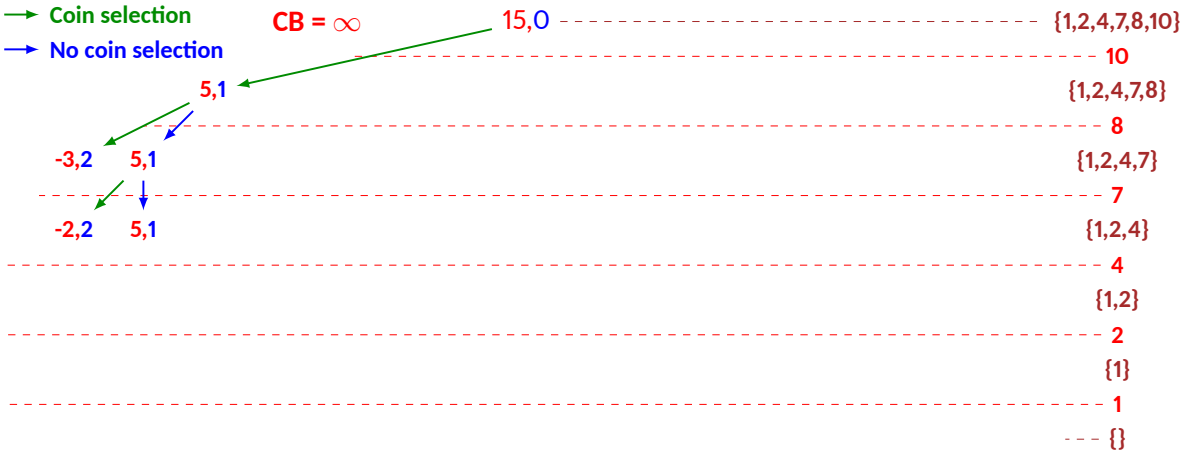
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



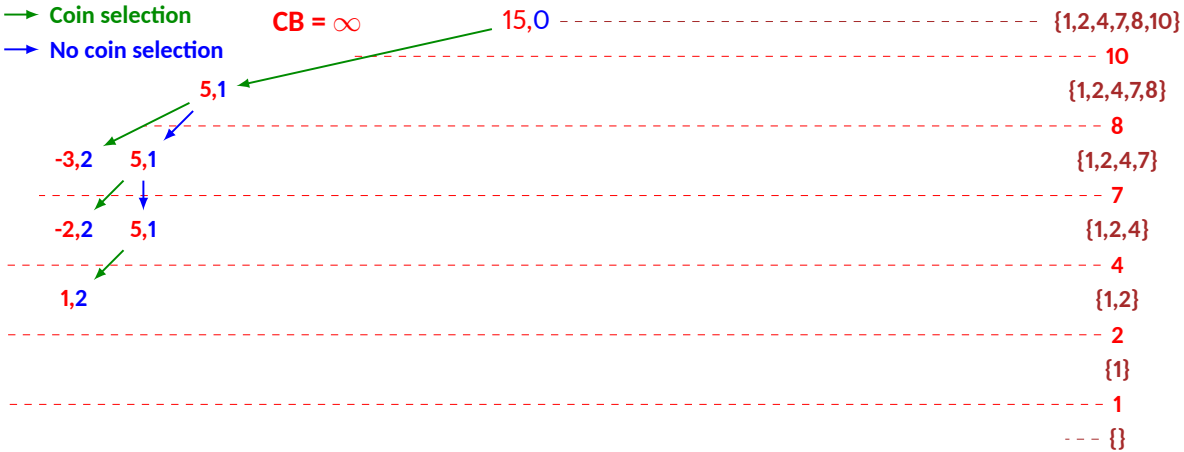
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



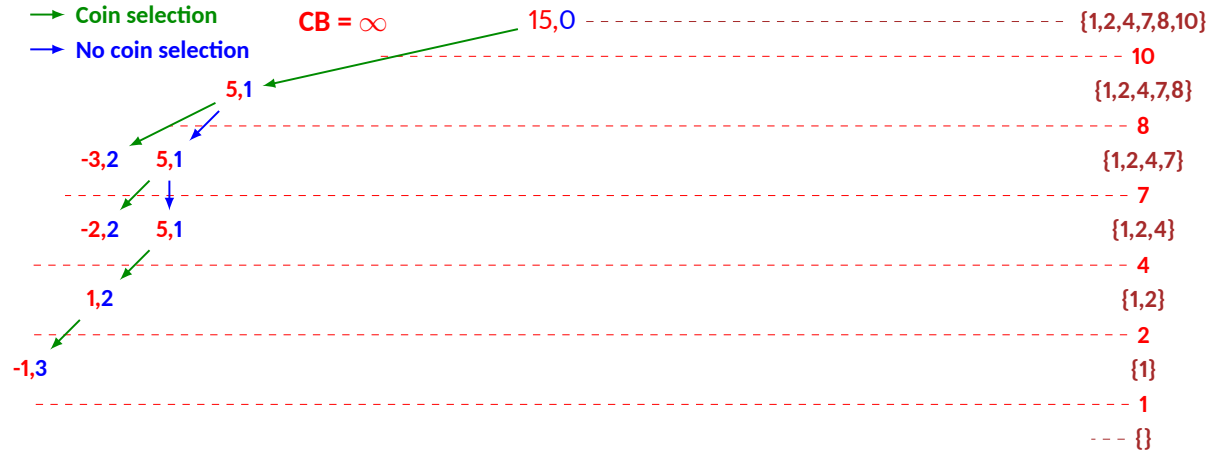
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



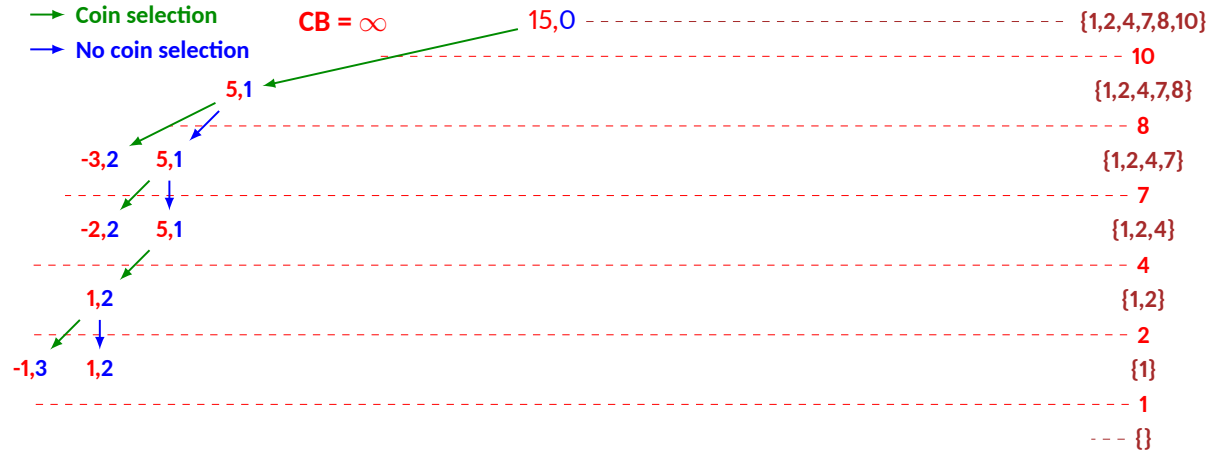
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



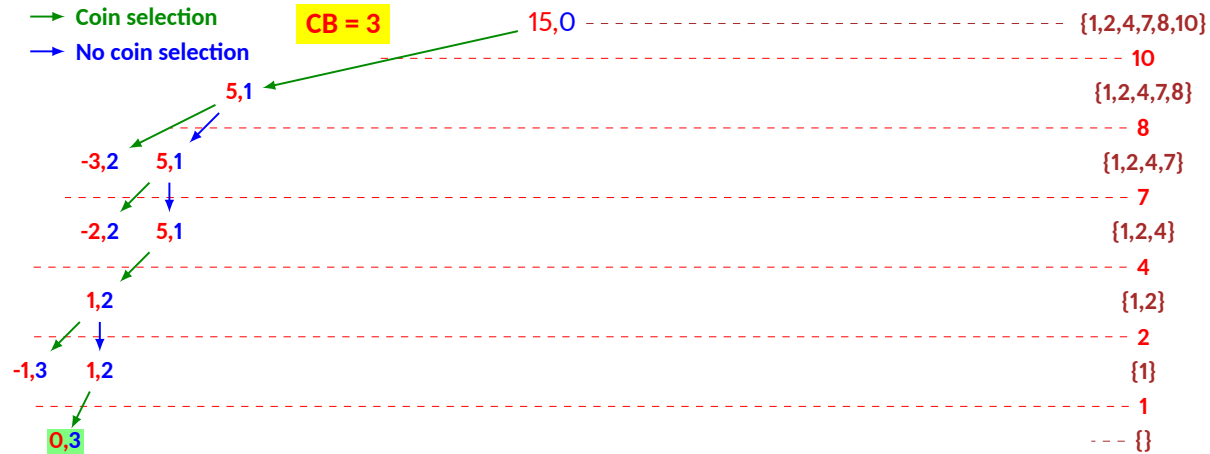
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



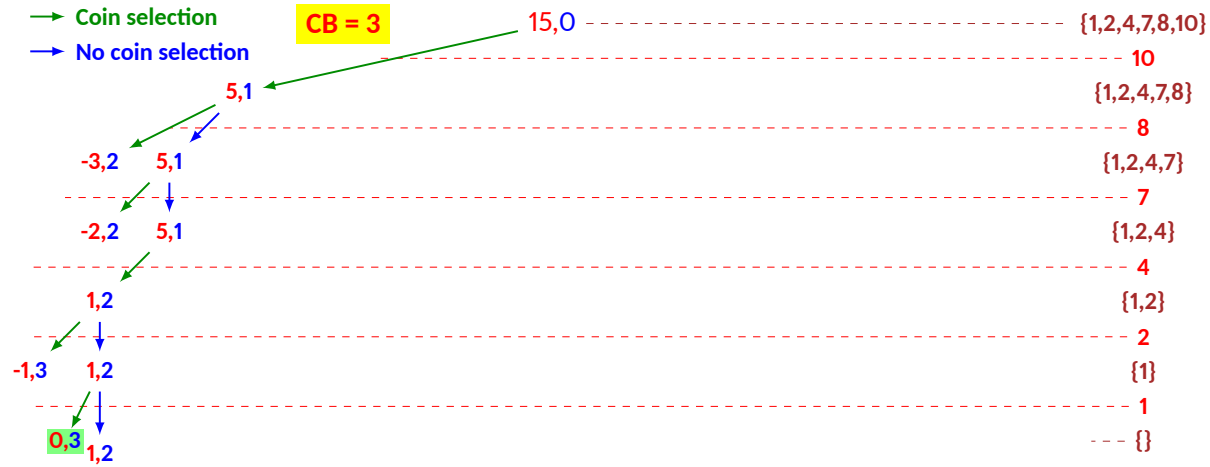
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



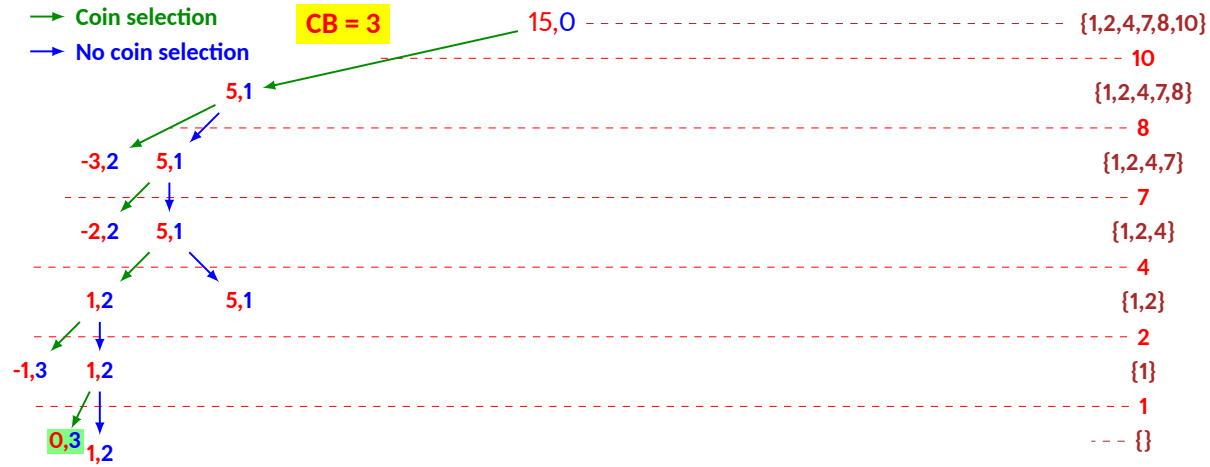
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



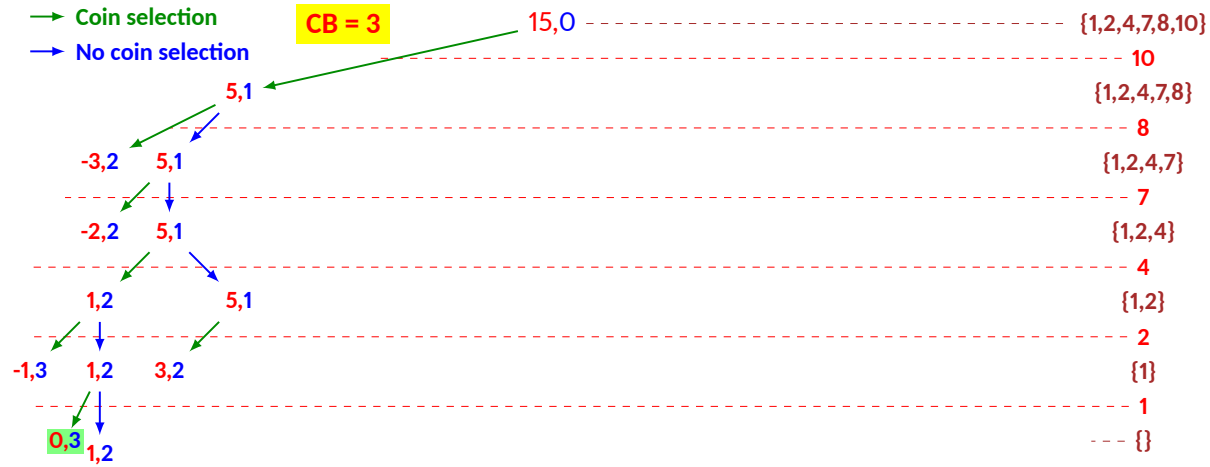
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



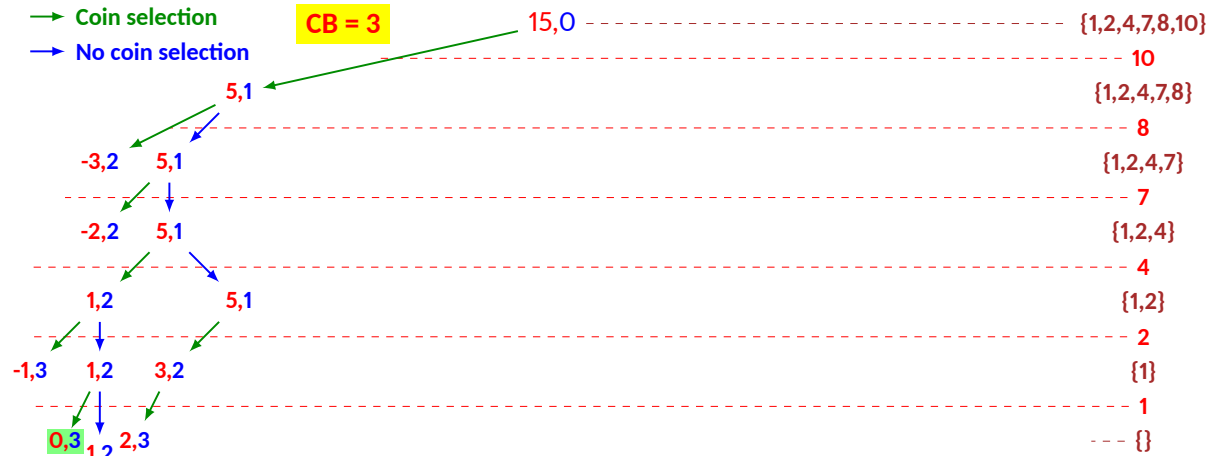
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



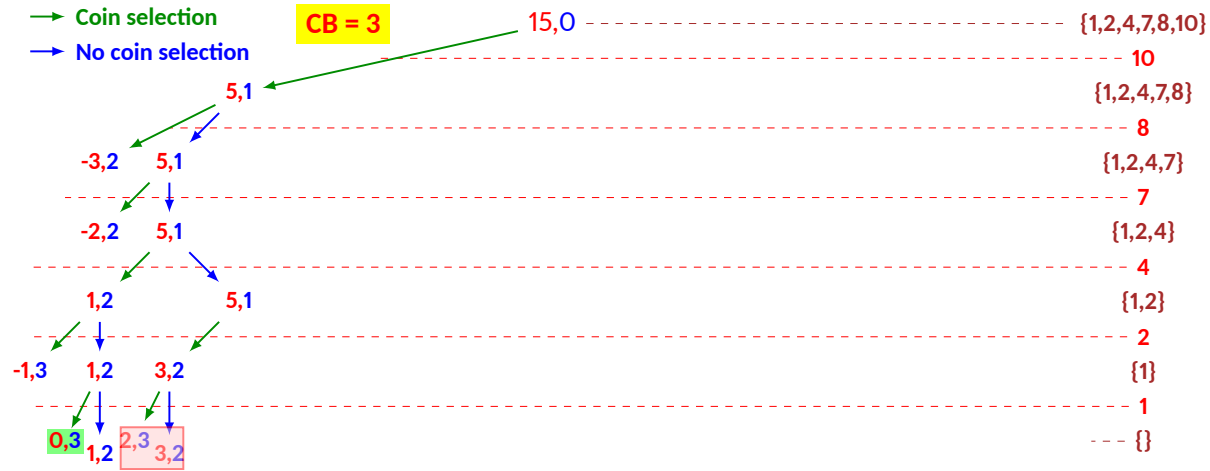
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



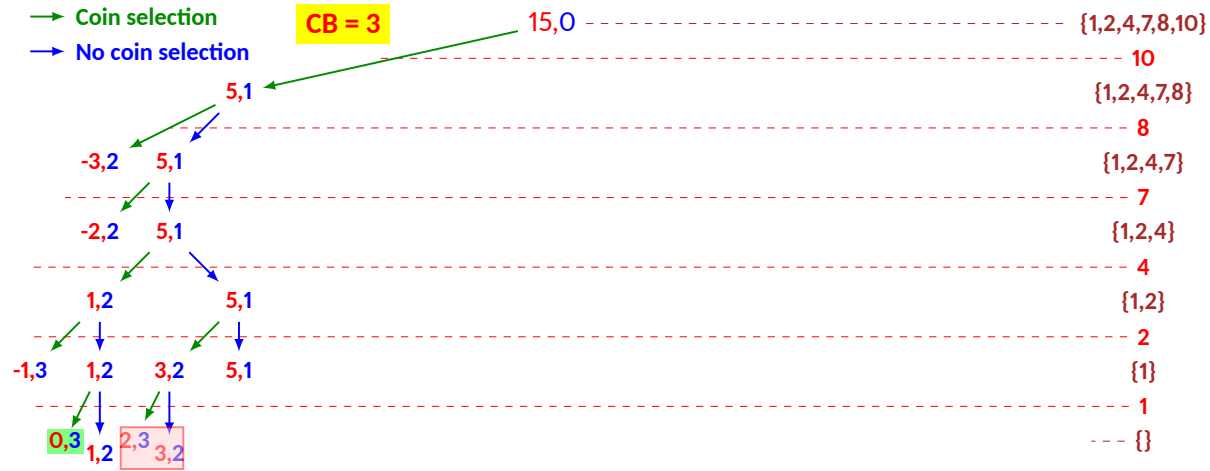
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



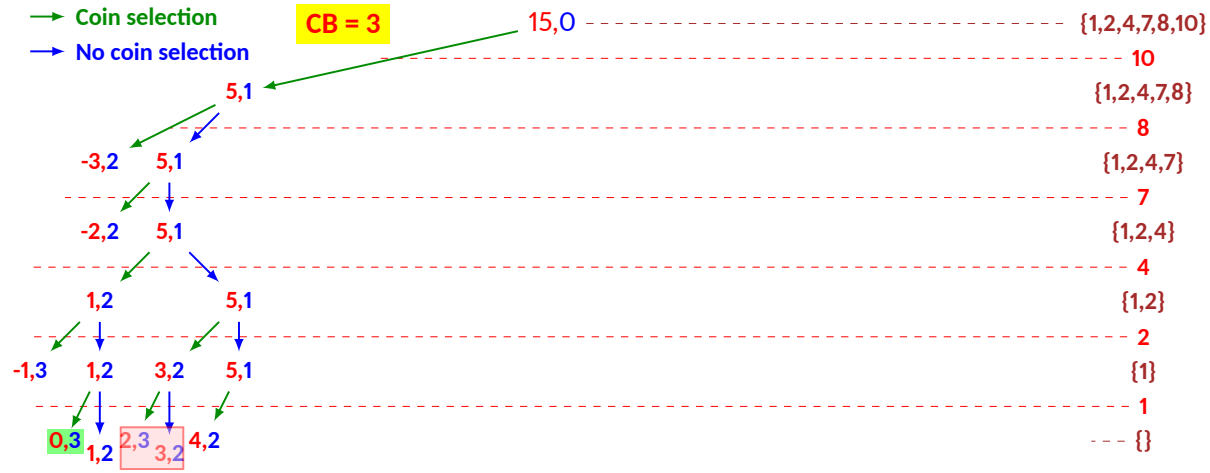
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



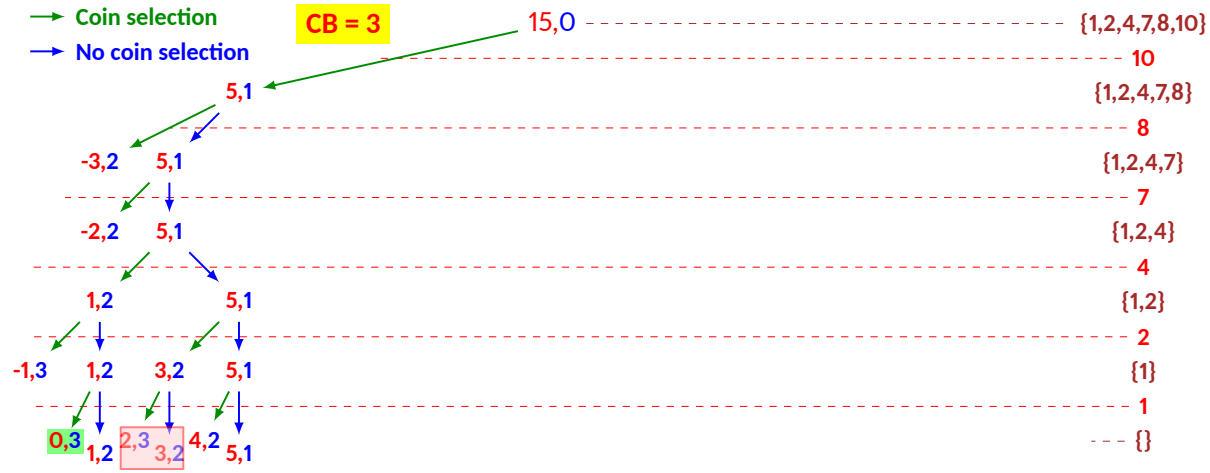
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



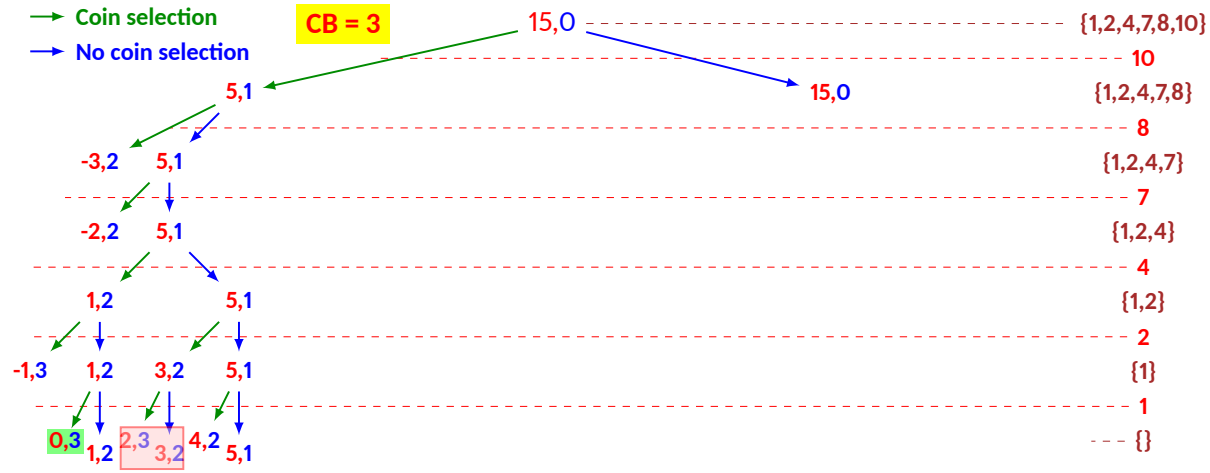
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



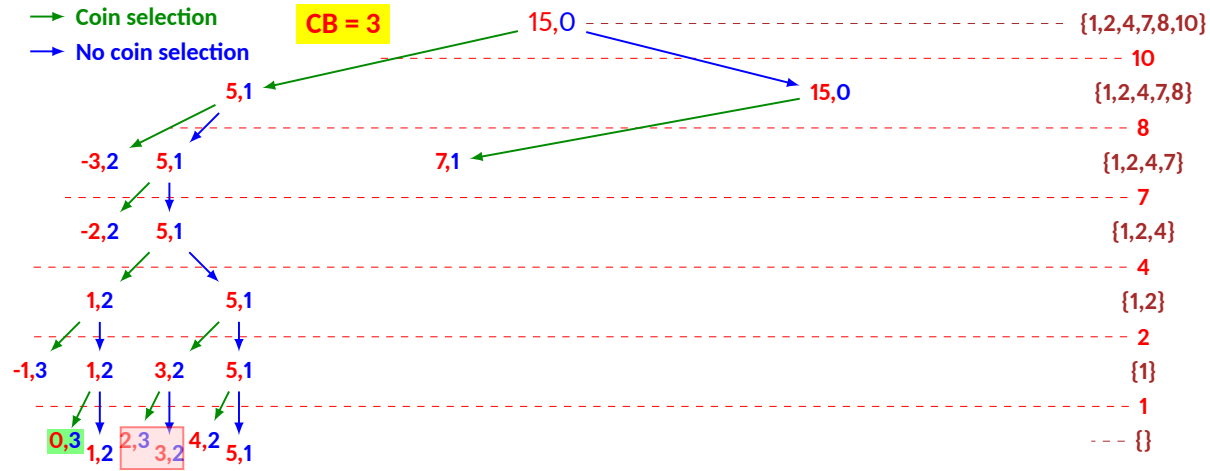
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



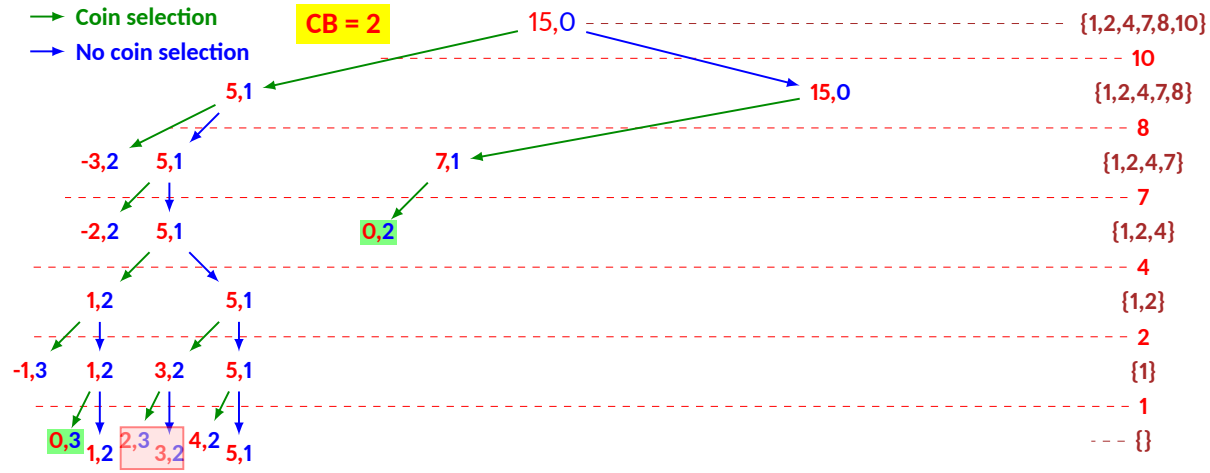
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



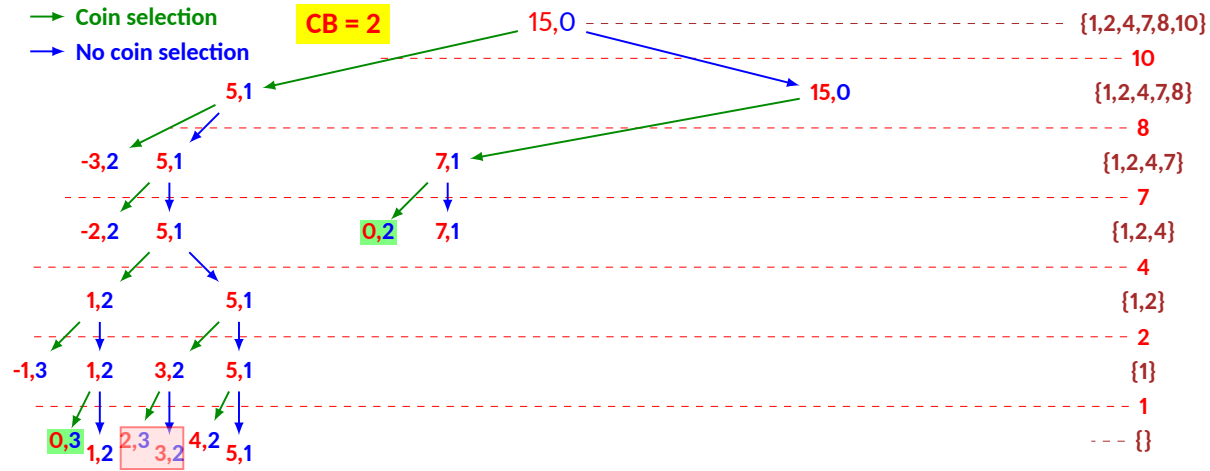
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



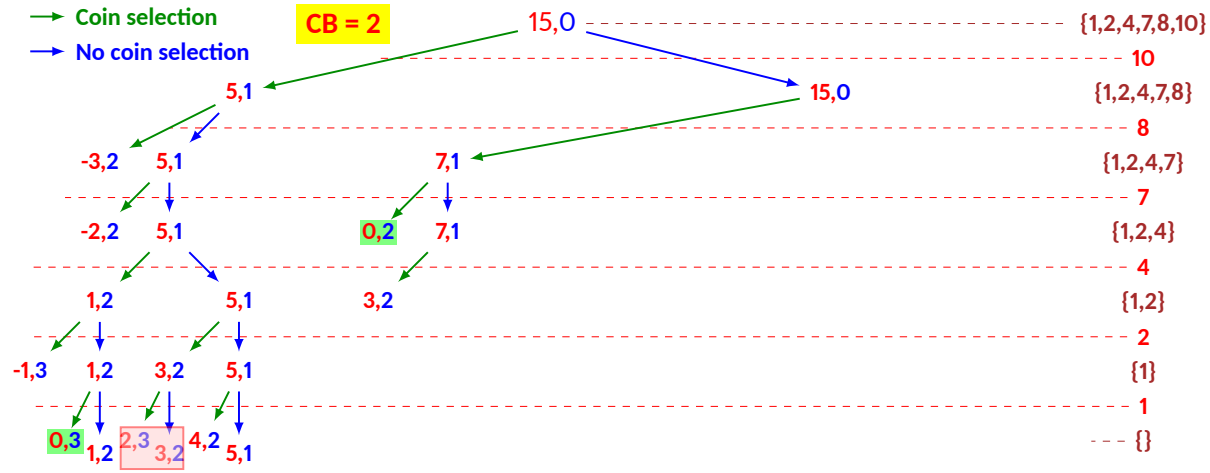
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



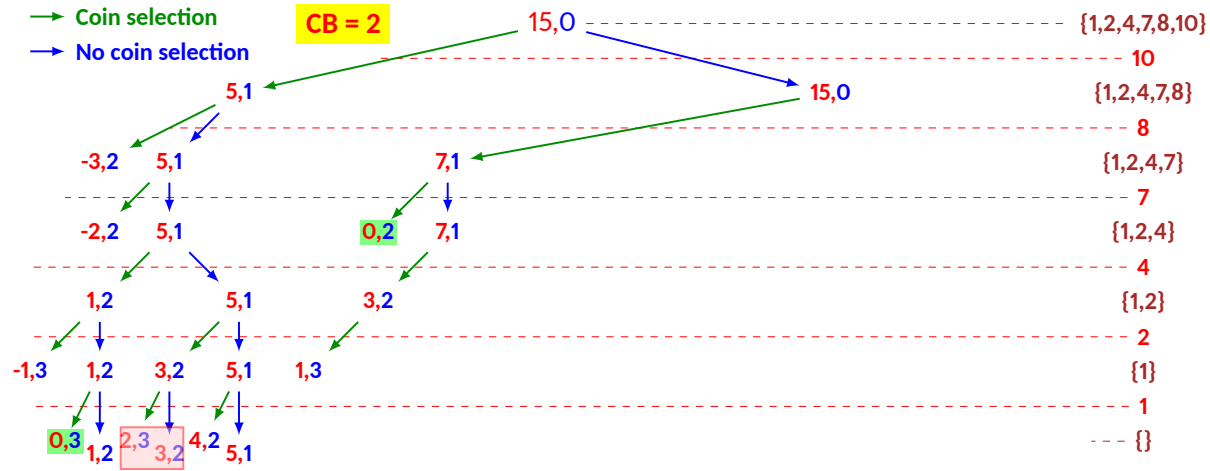
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



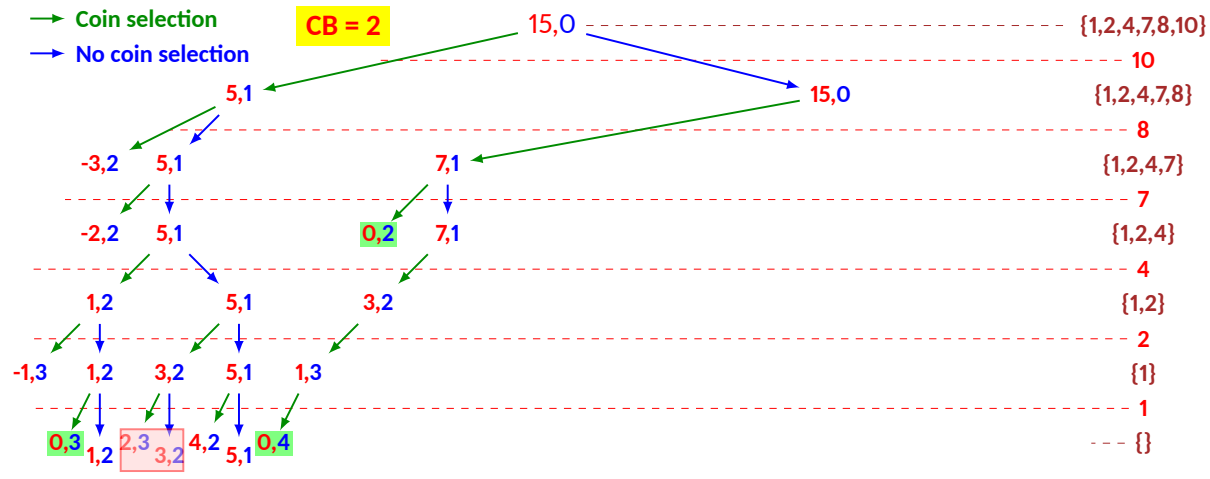
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



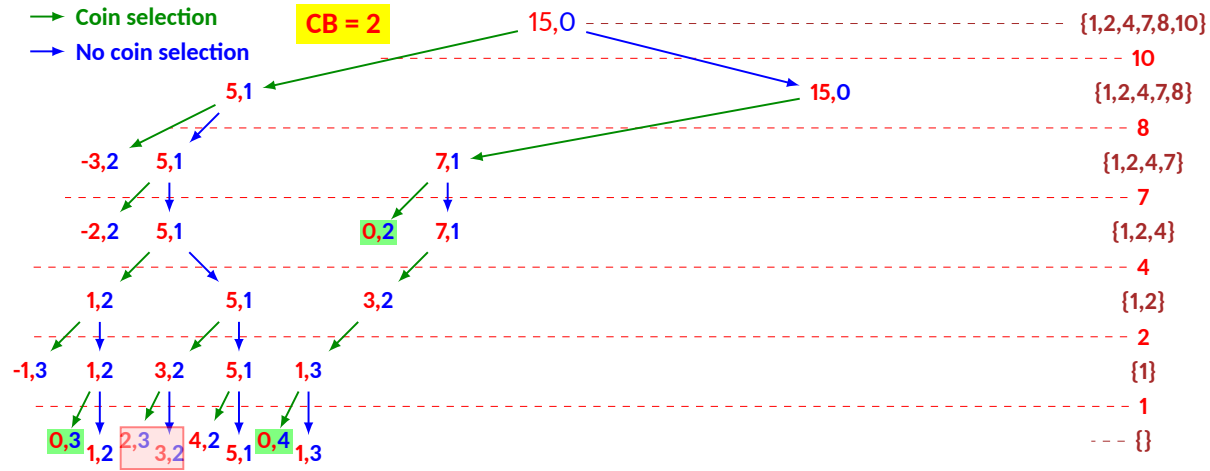
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



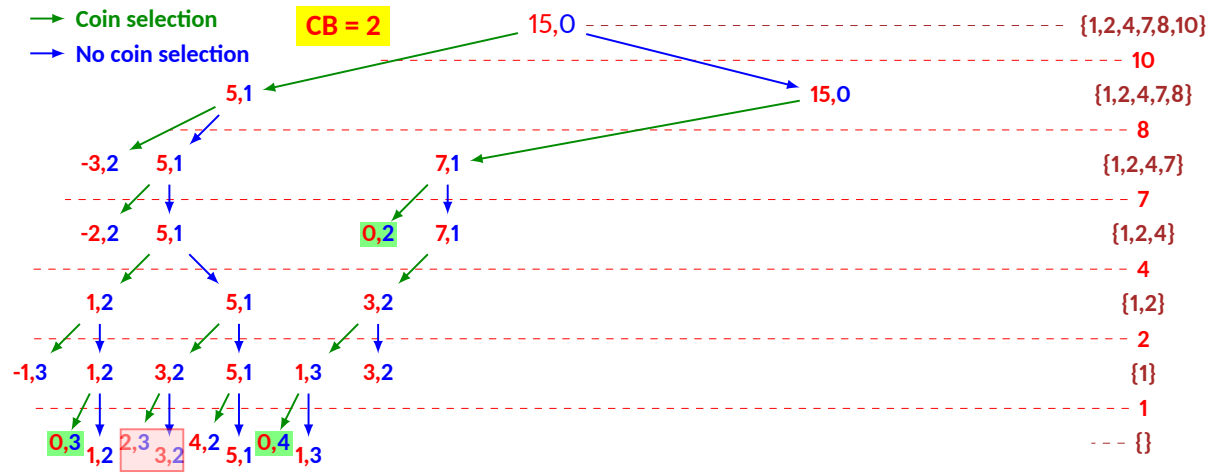
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



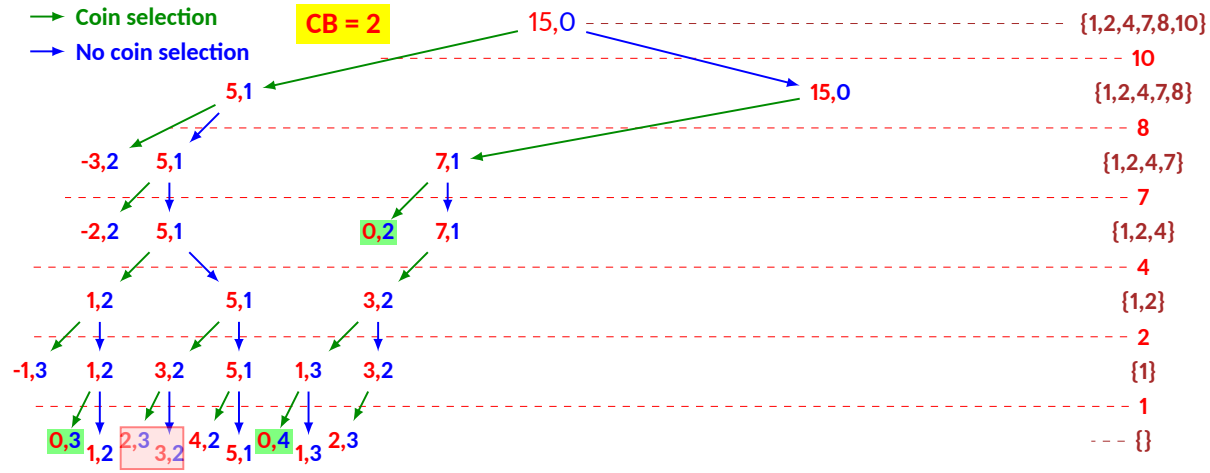
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



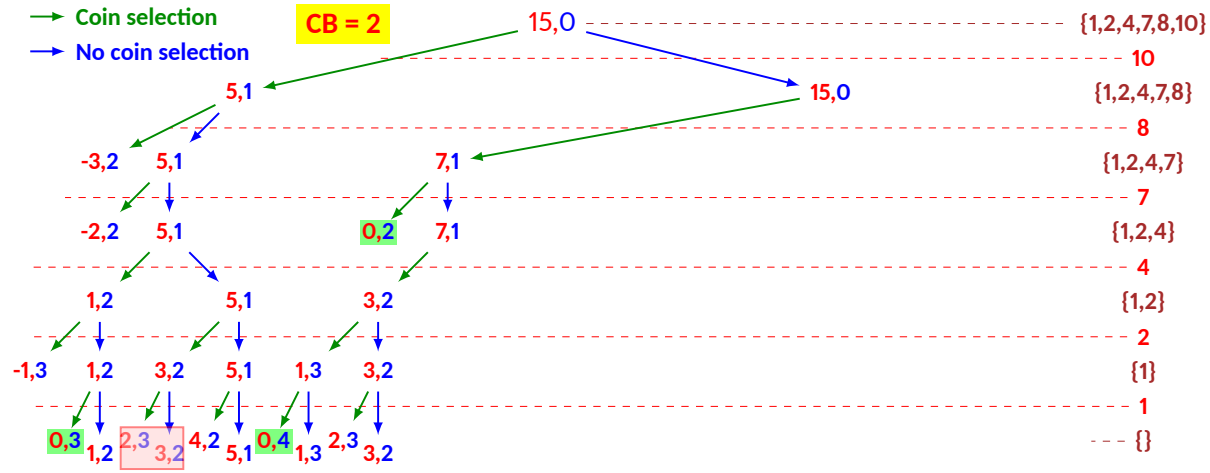
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



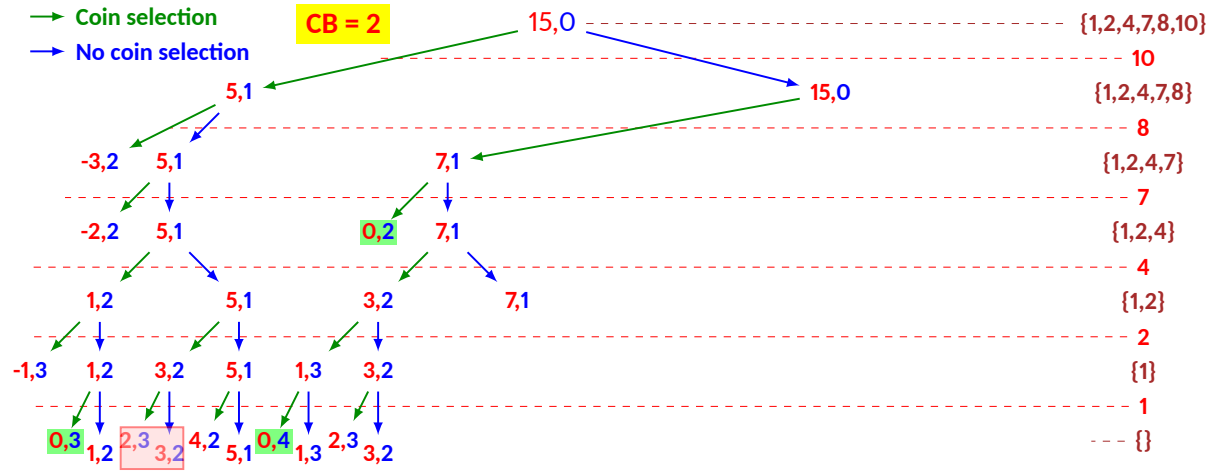
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



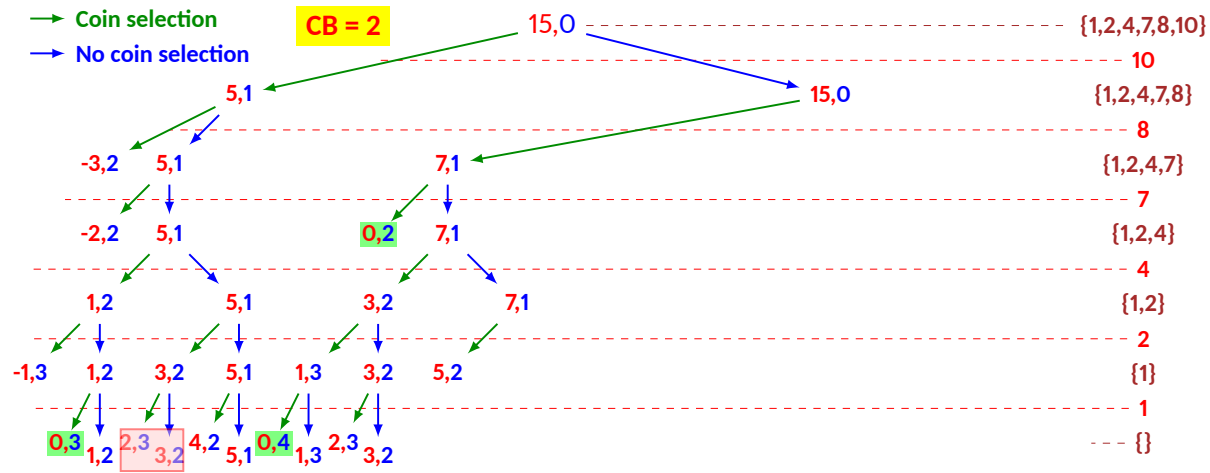
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



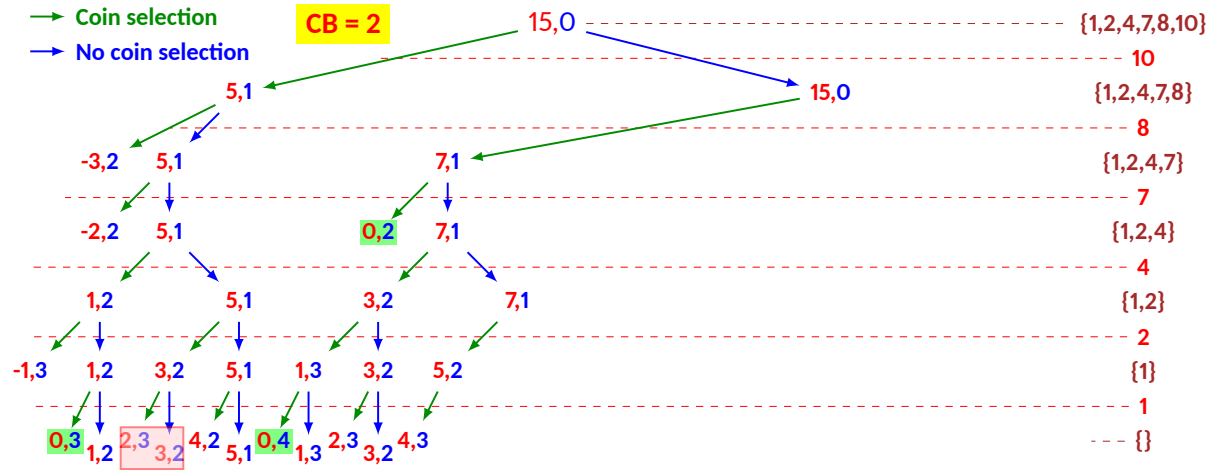
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



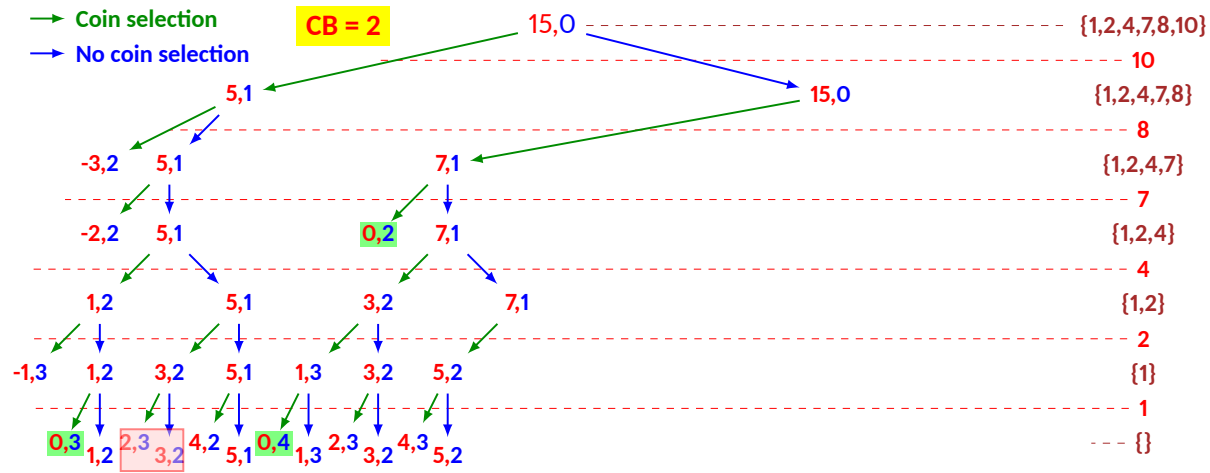
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



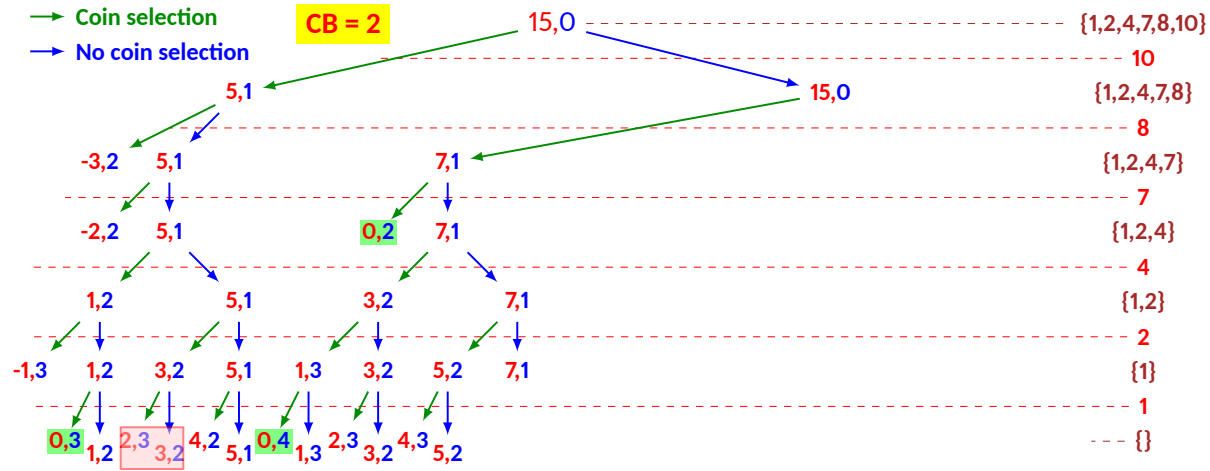
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



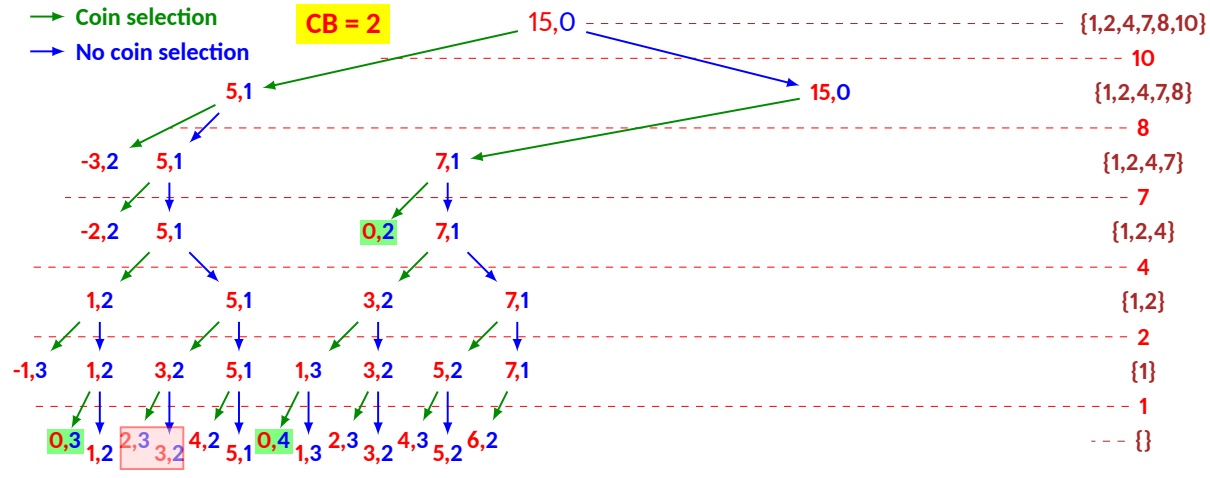
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



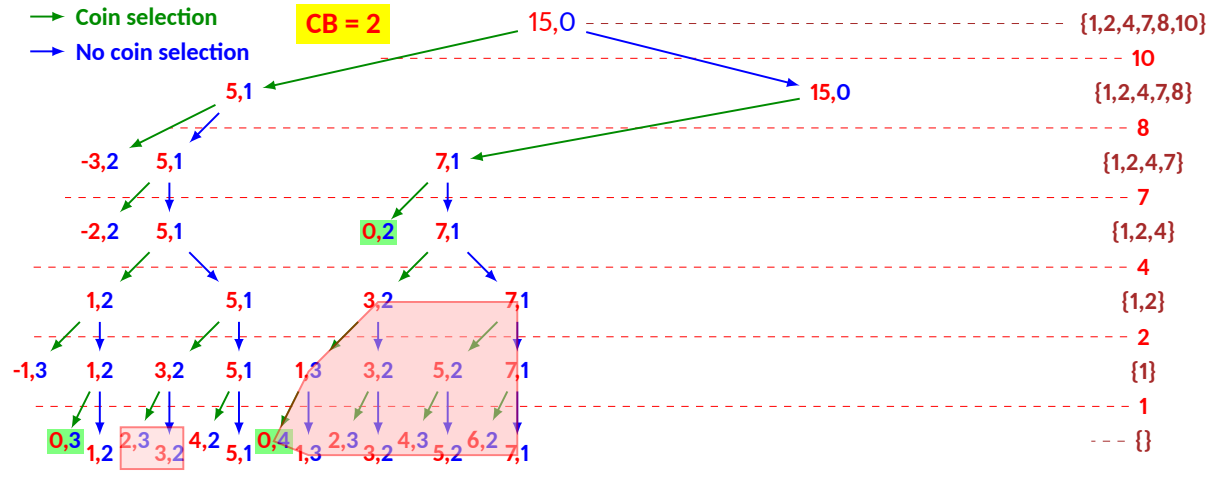
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



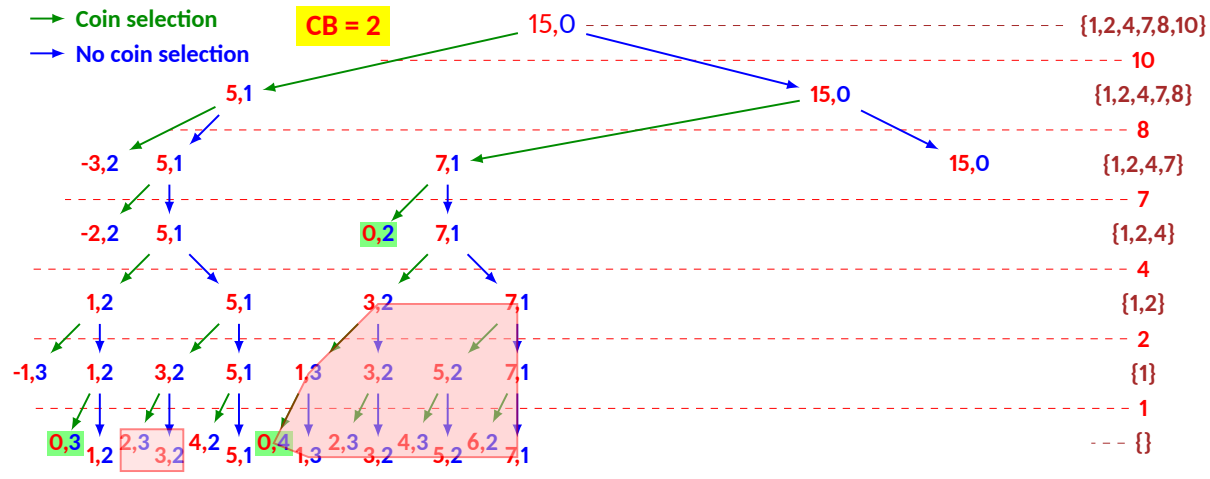
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



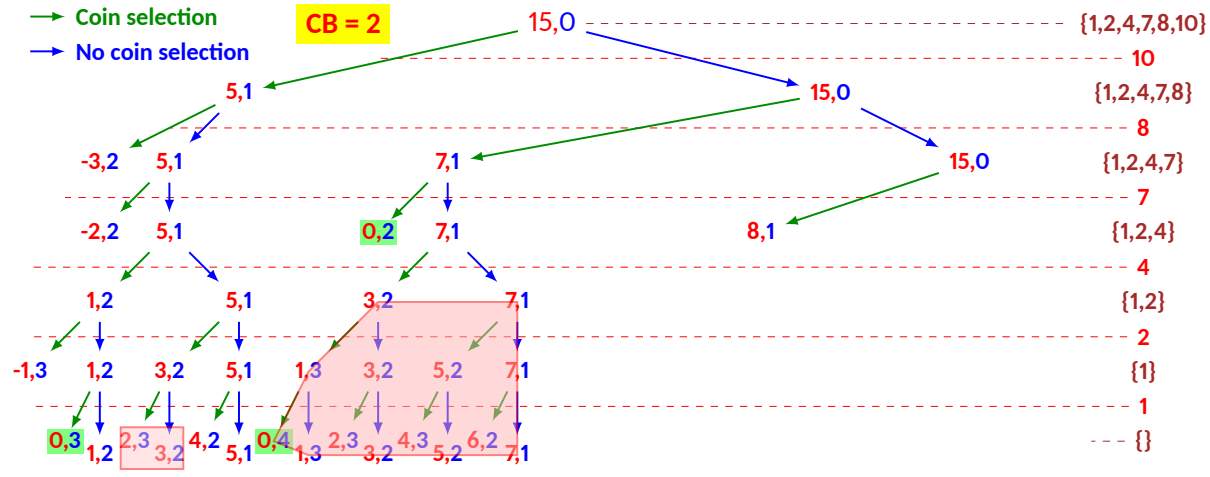
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



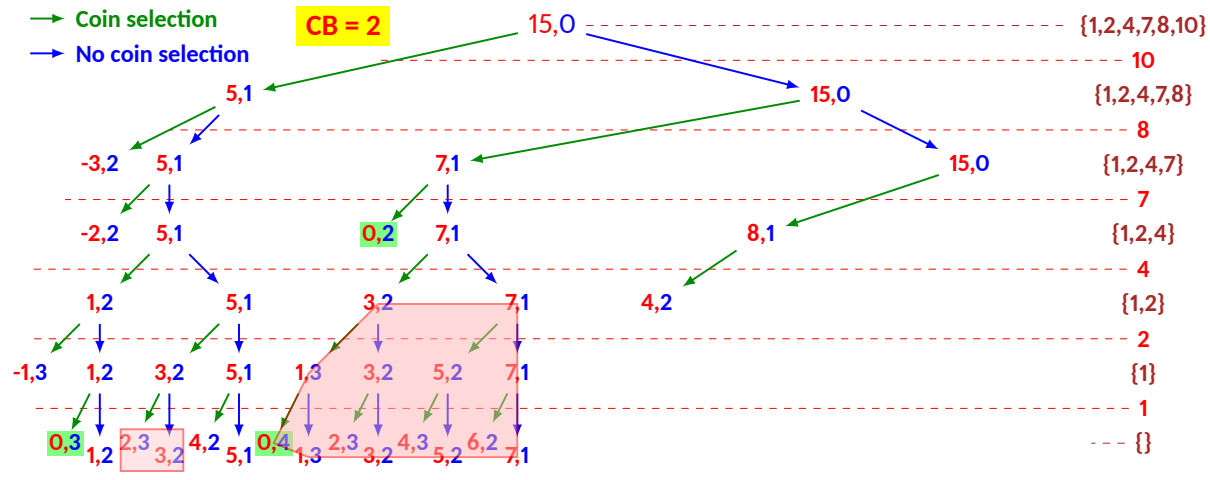
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



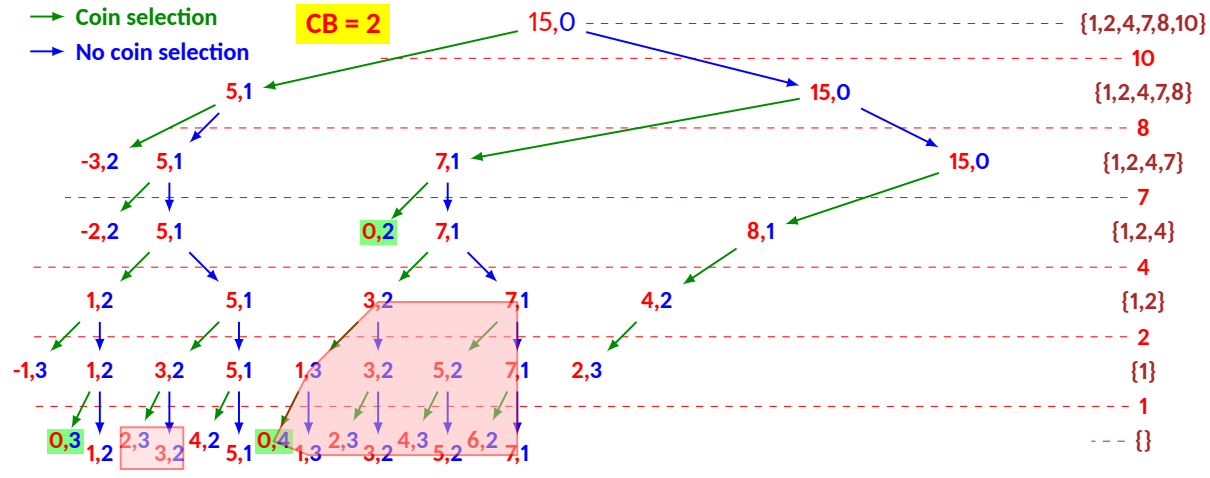
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



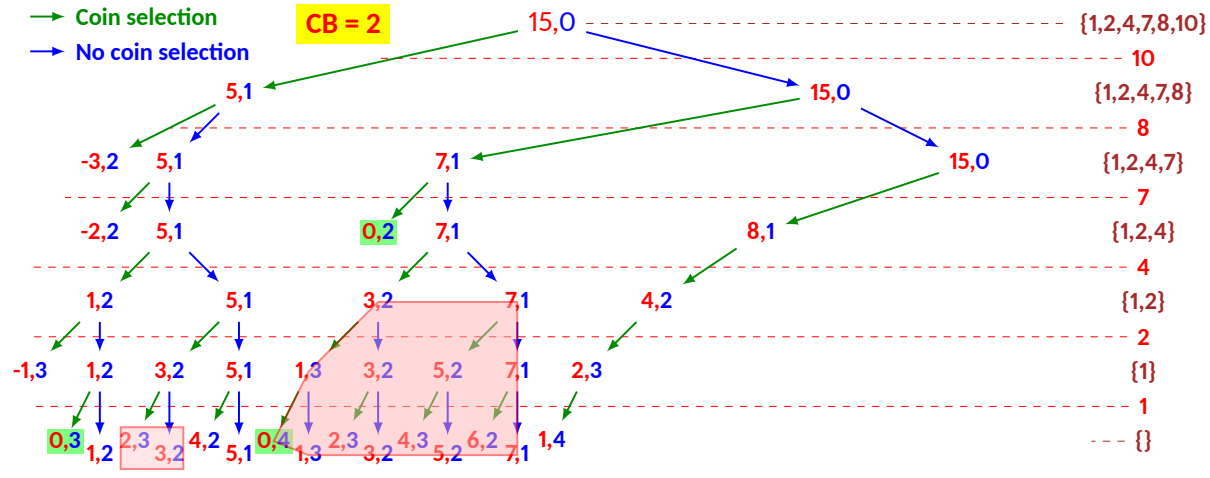
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



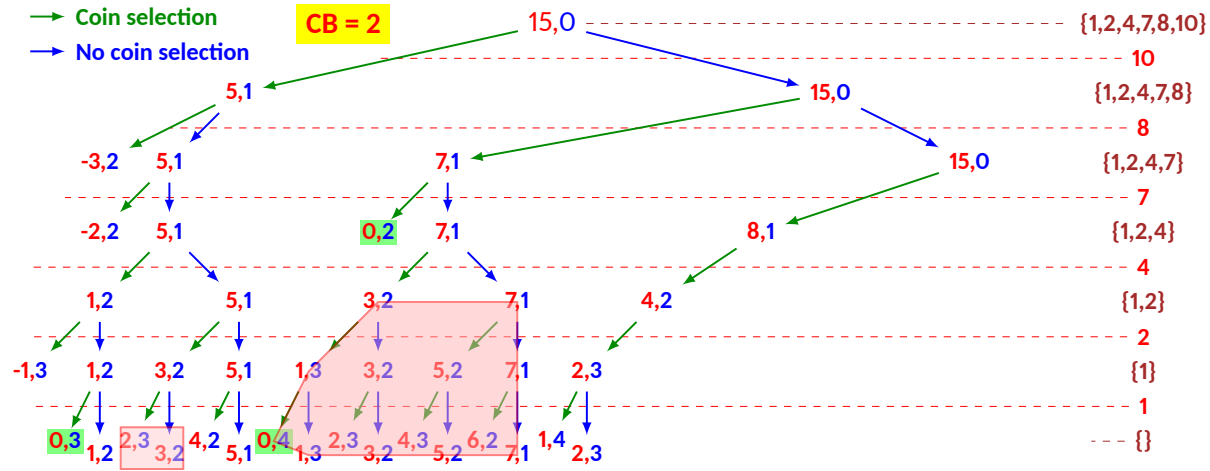
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



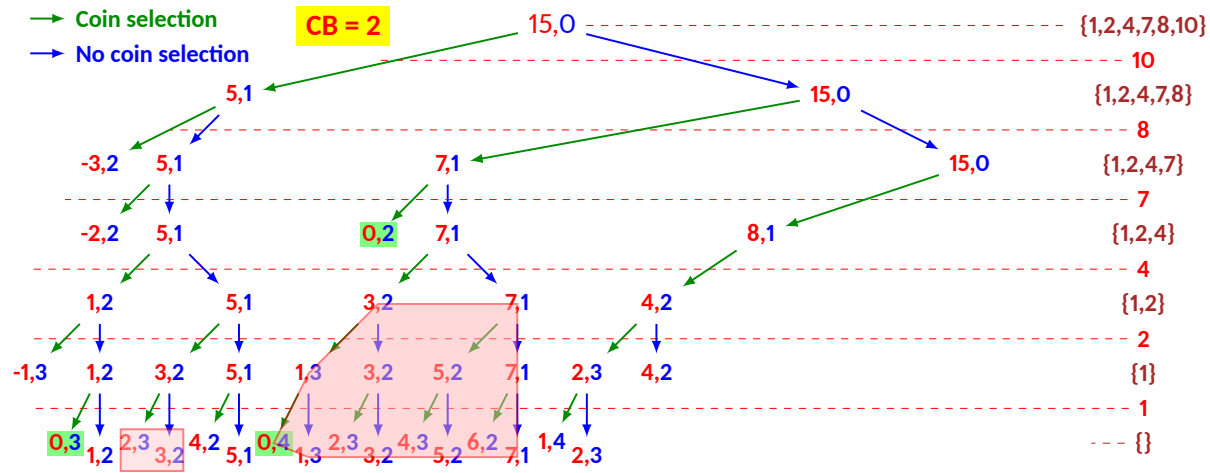
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



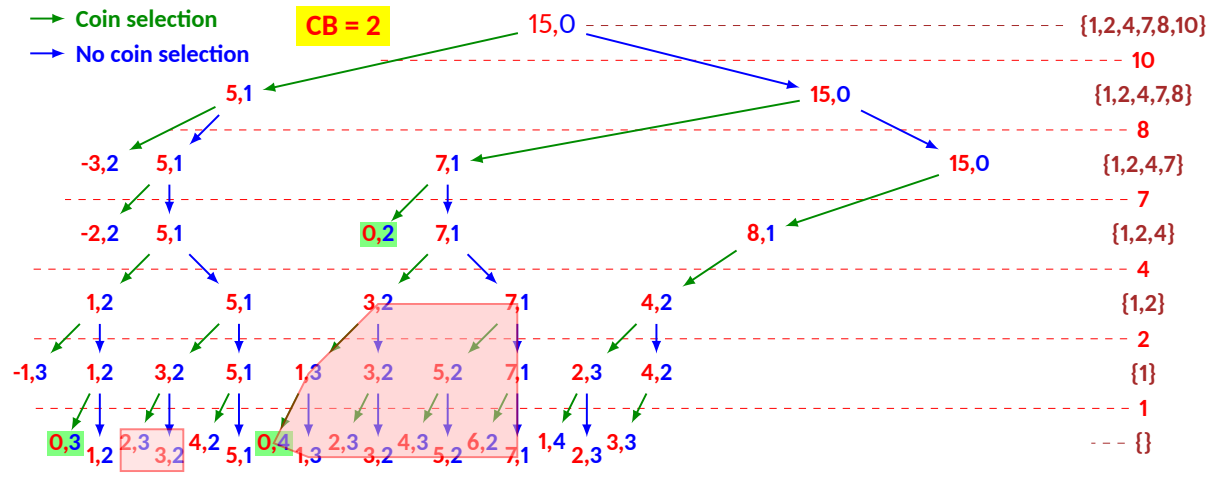
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



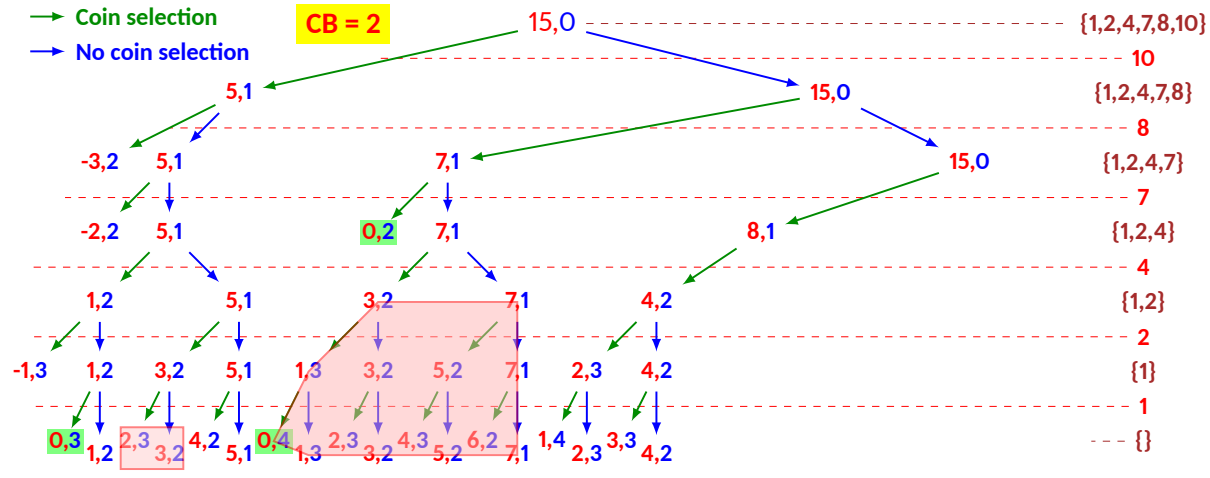
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



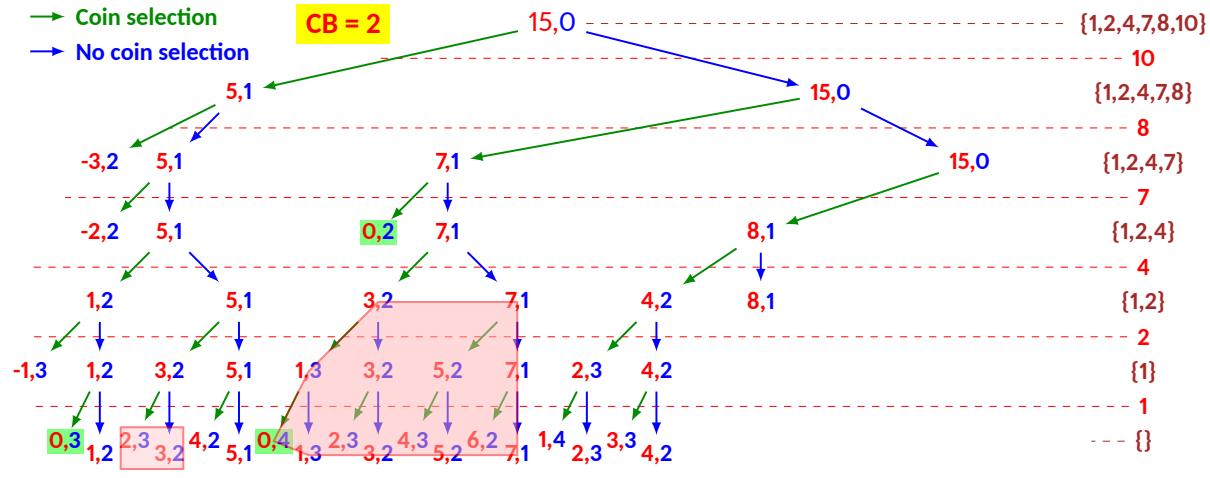
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



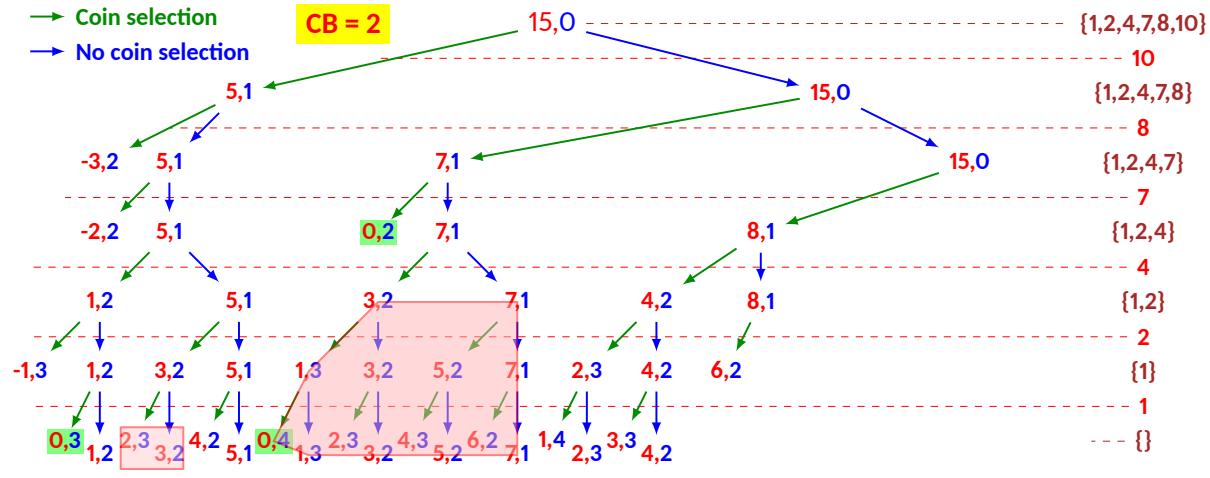
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



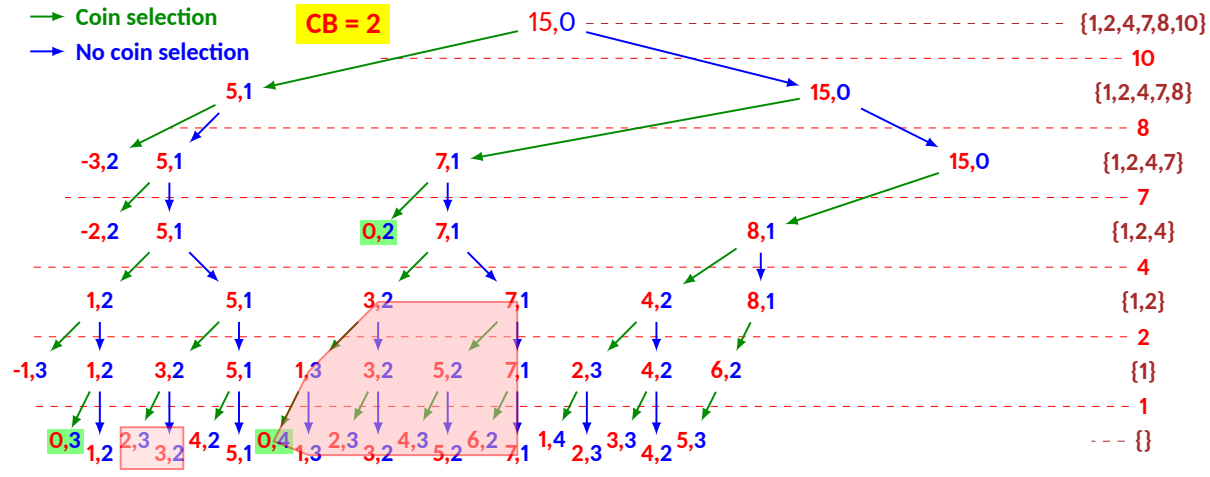
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



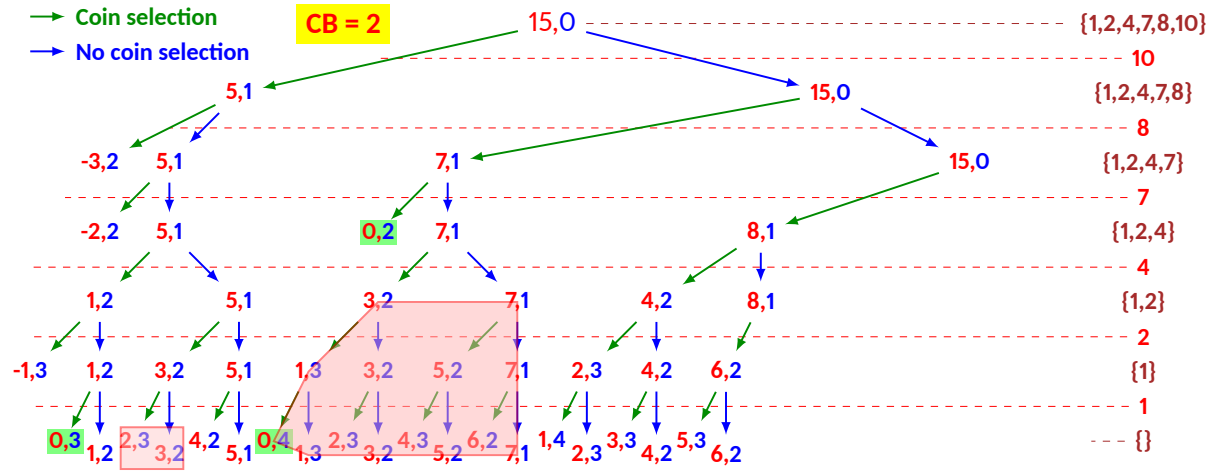
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



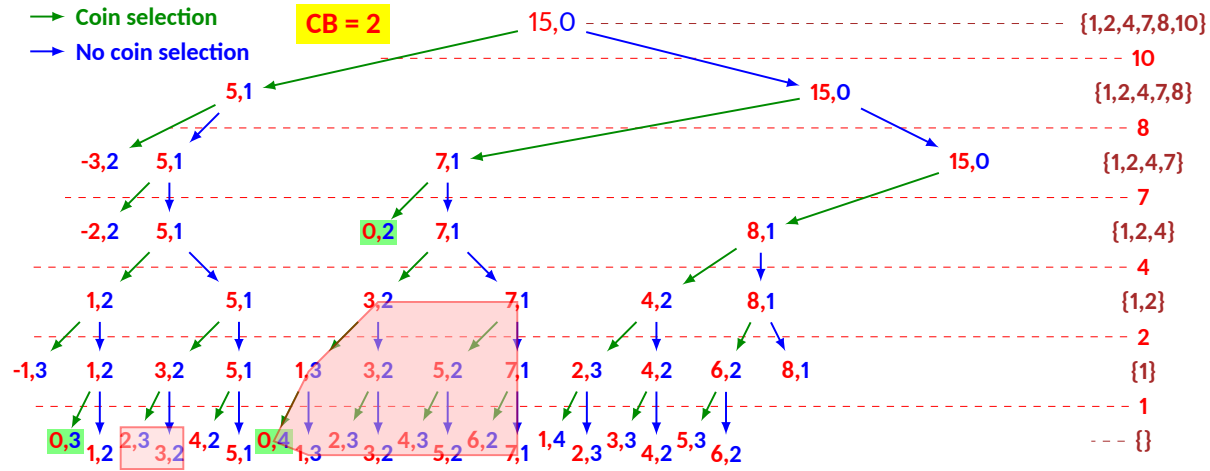
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



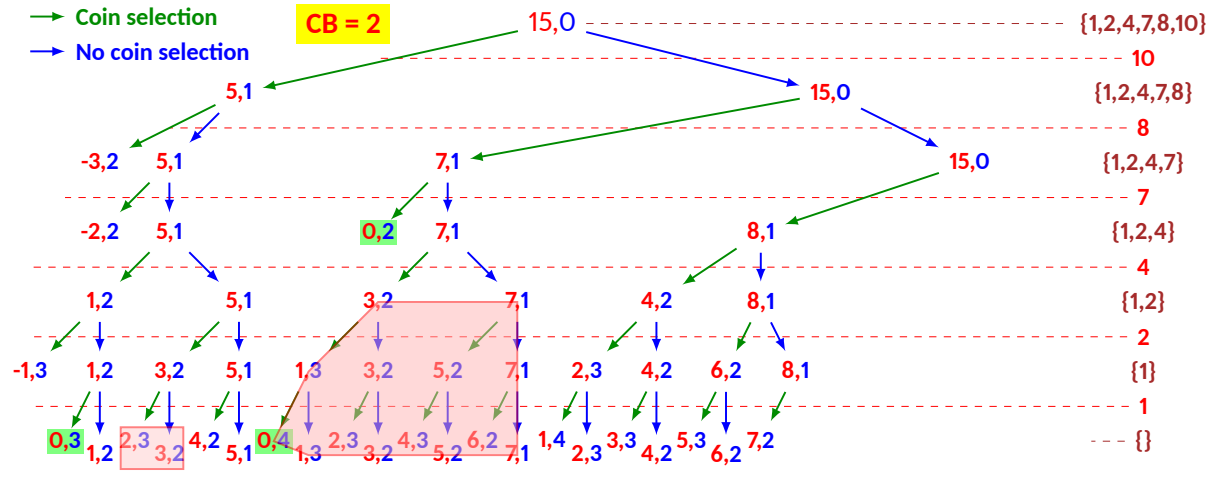
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



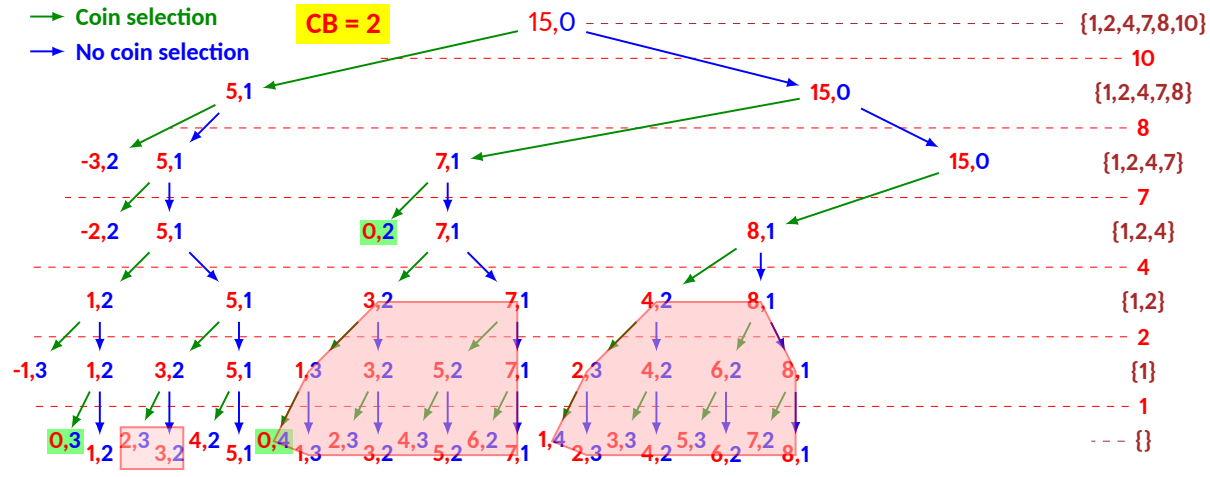
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



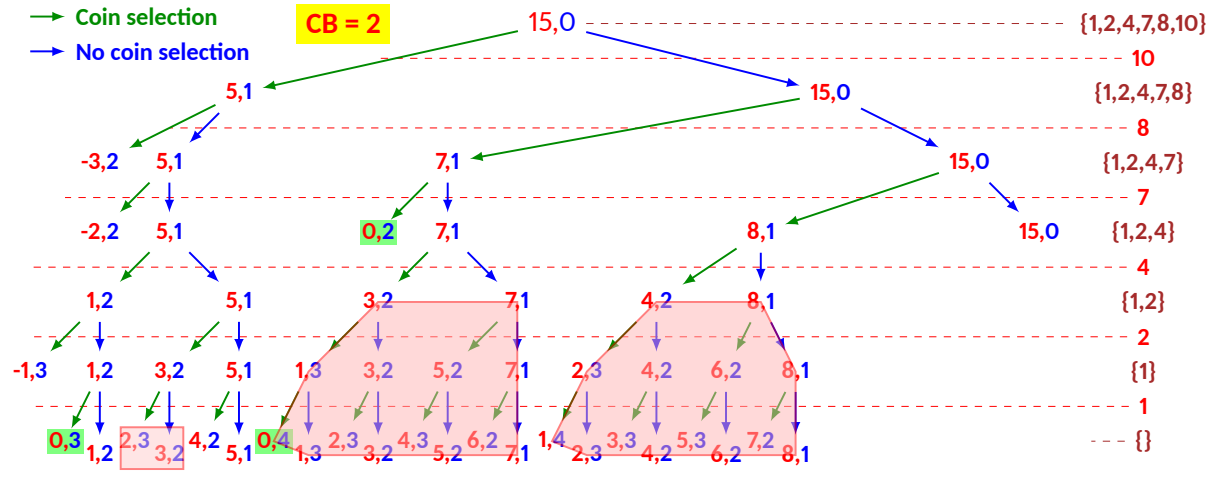
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



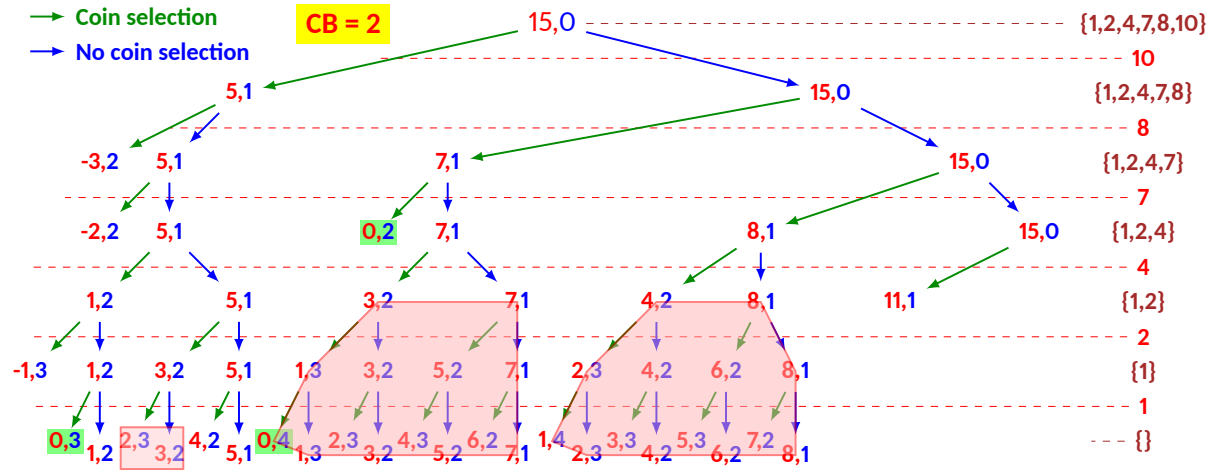
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



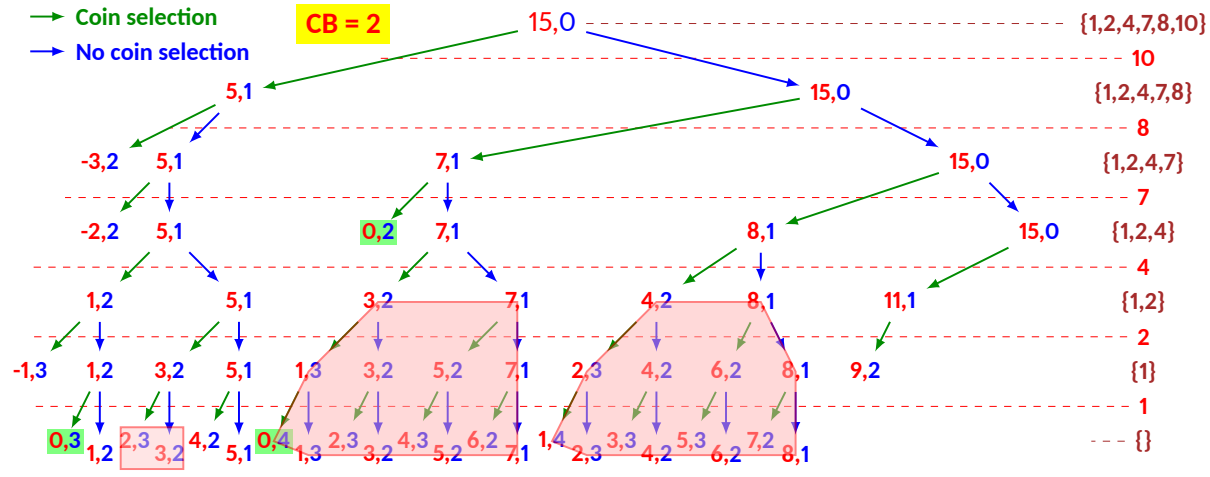
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



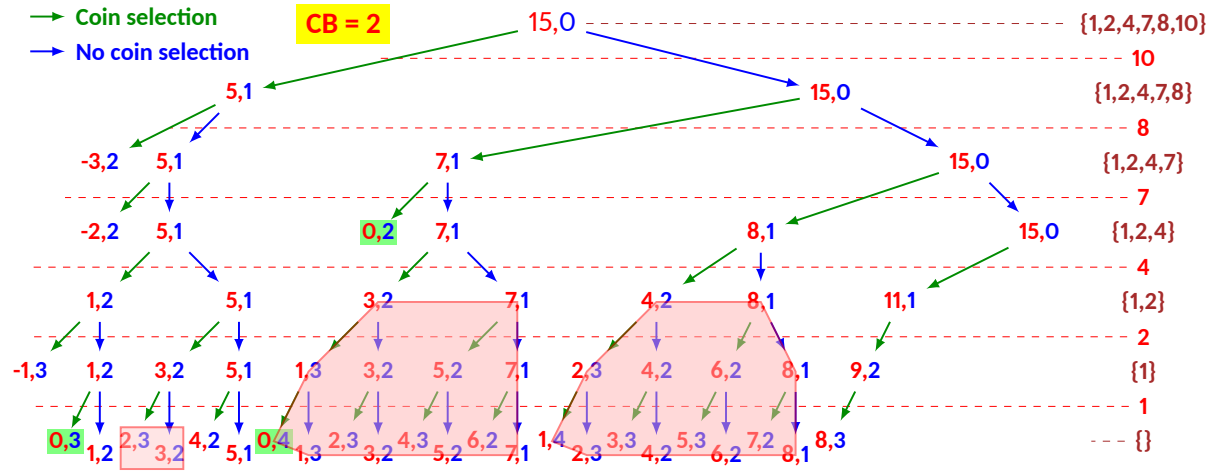
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



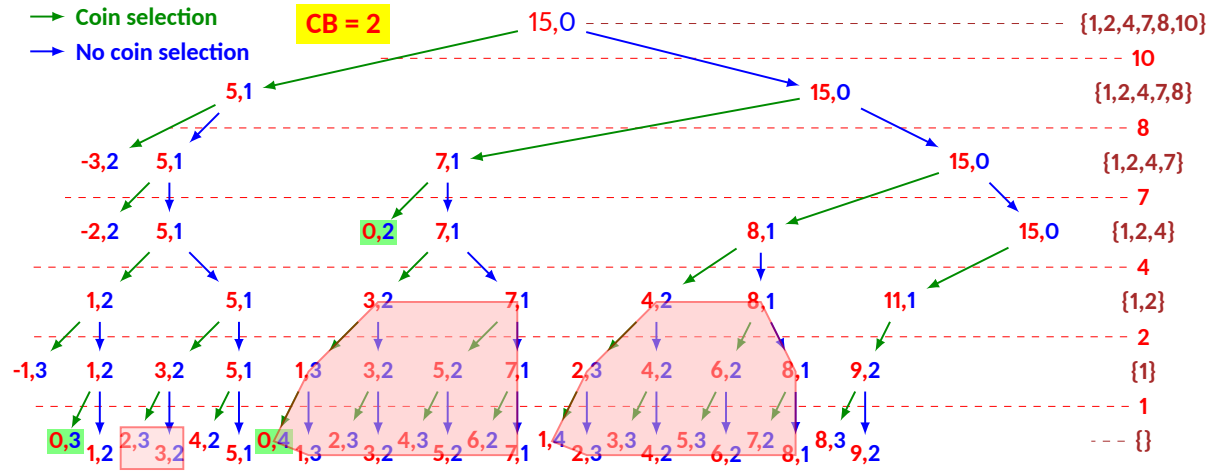
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



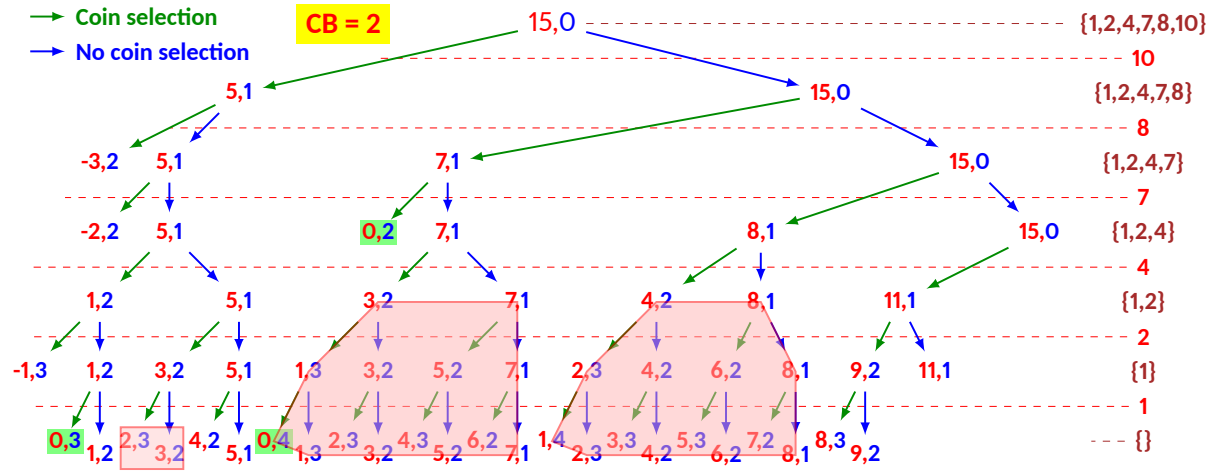
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



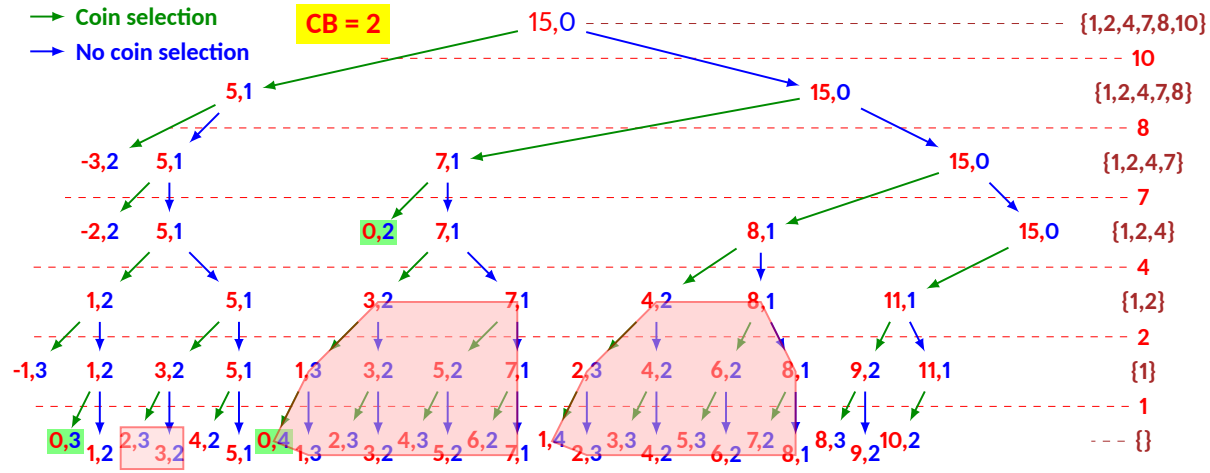
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



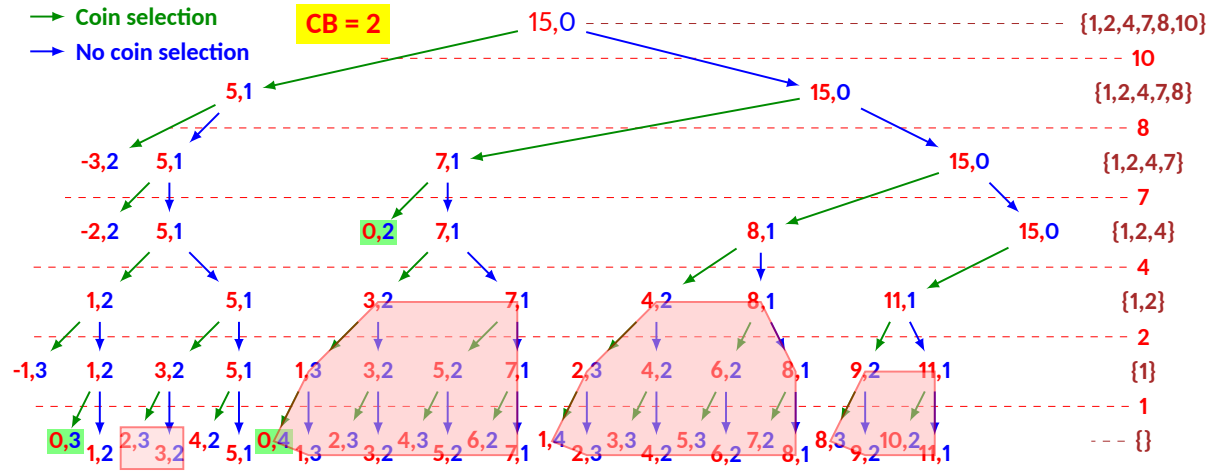
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



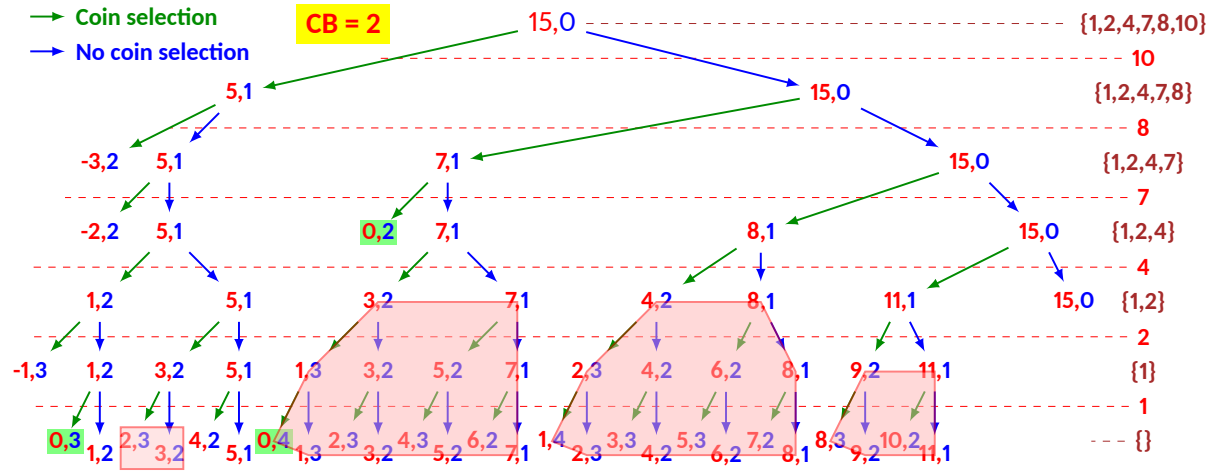
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



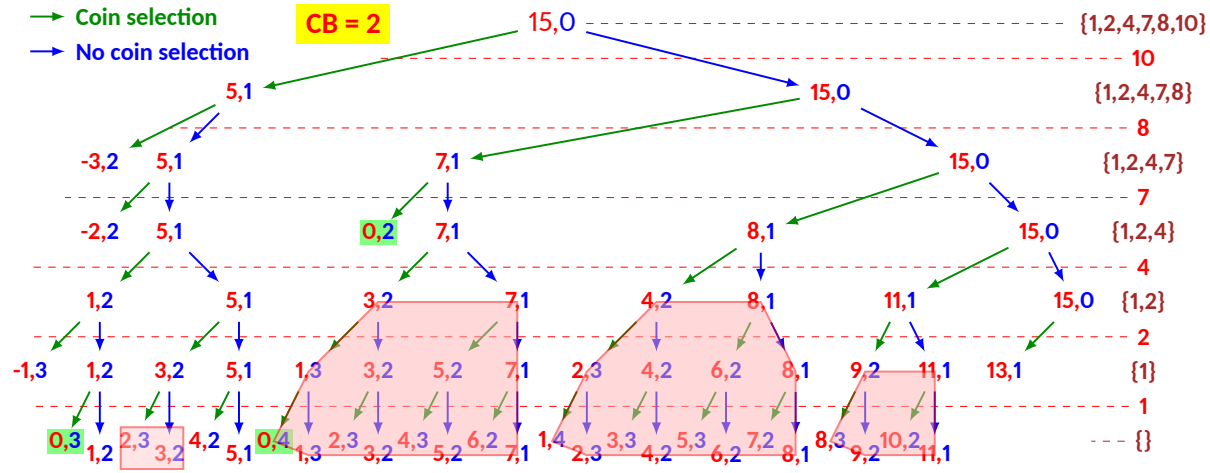
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



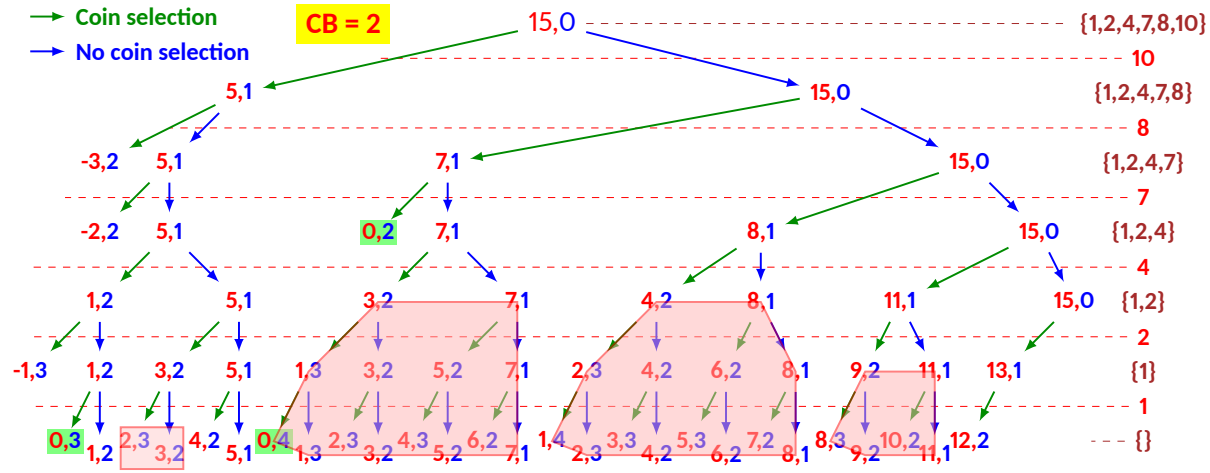
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



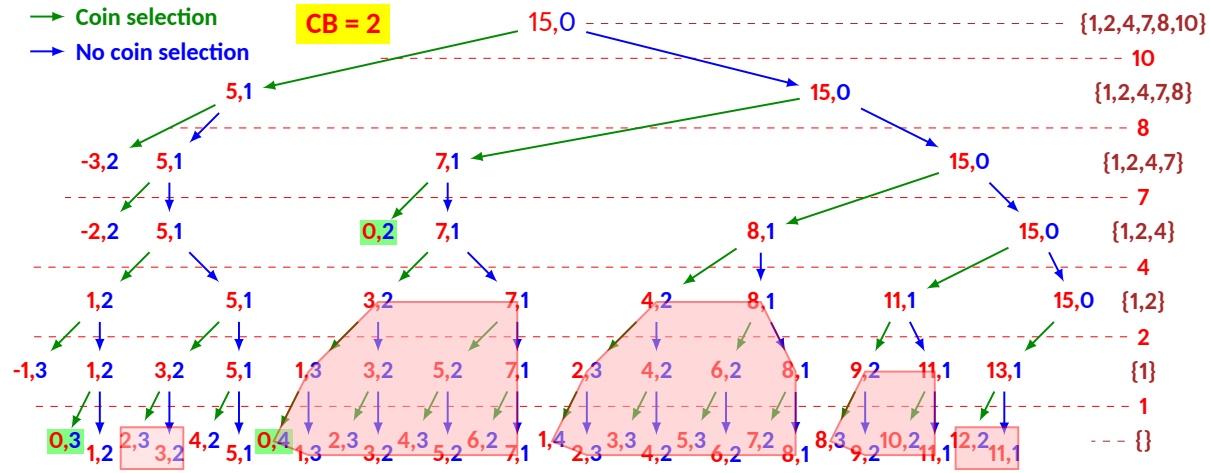
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



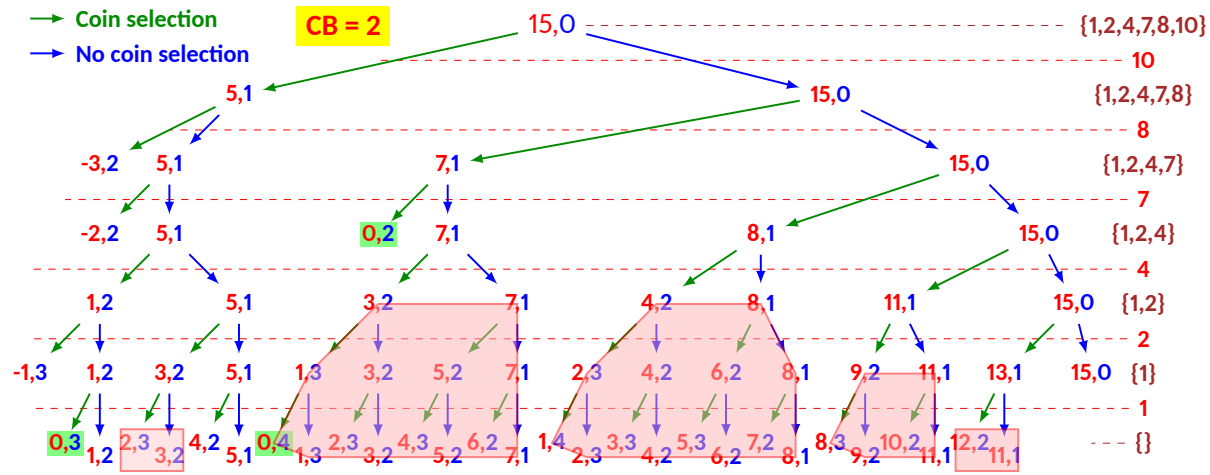
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



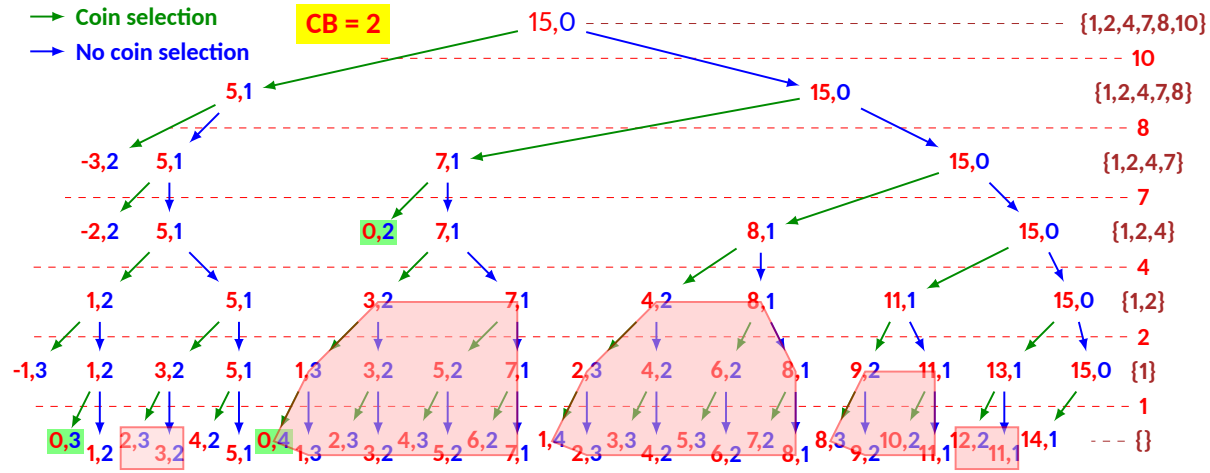
Coin change: DFBB

- Given a set of coins C, what is the minimum number coins required to provide sum S?
- Example: $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



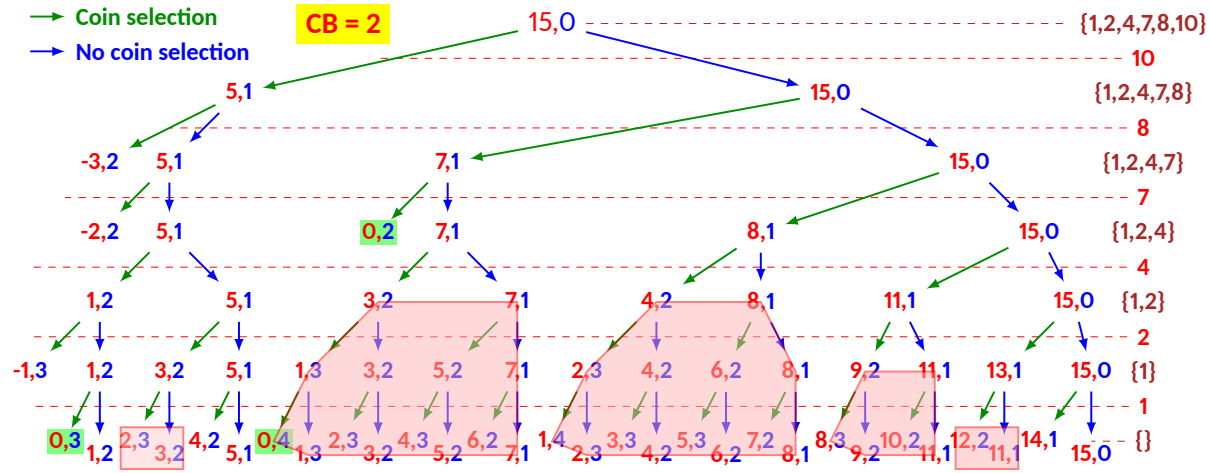
Coin change: DFBB

- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



Coin change: DFBB

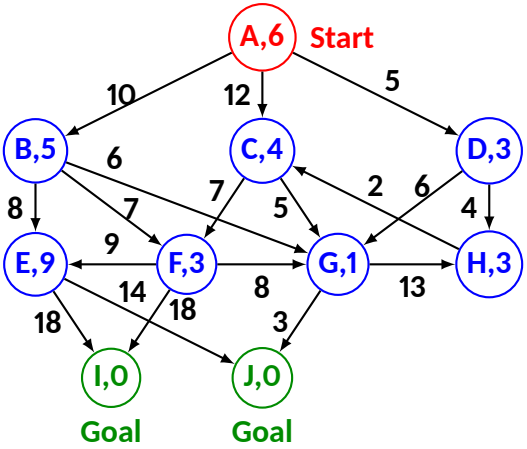
- **Given a set of coins C, what is the minimum number coins required to provide sum S?**
- **Example:** $C = \{1, 2, 4, 7, 8, 10\}, S = 15$



Iterative deepening A* (IDA*)

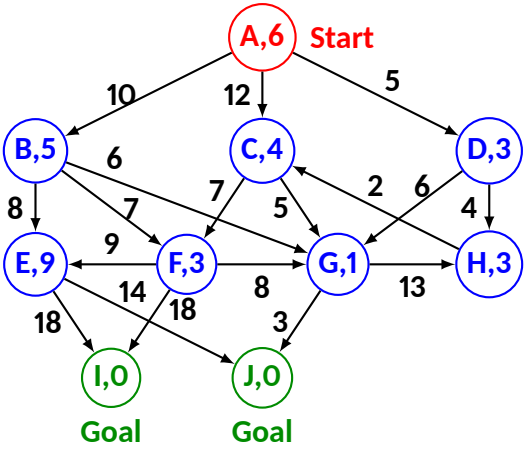
1. **Set Cut-off Bound to $f(s)$**
2. **Perform DFBB with Cut-off Bound. Backtrack from any node whose $f(n) >$ Cut-off Bound.**
3. **If Solution is Found, at the end of one Iteration, Terminate with Solution**
4. **If Solution is not found in any iteration, then update Cut-off Bound to the lowest $f(n)$ among all nodes from which the algorithm Backtracked.**
5. **Go to Step 2**

Execution of IDA*



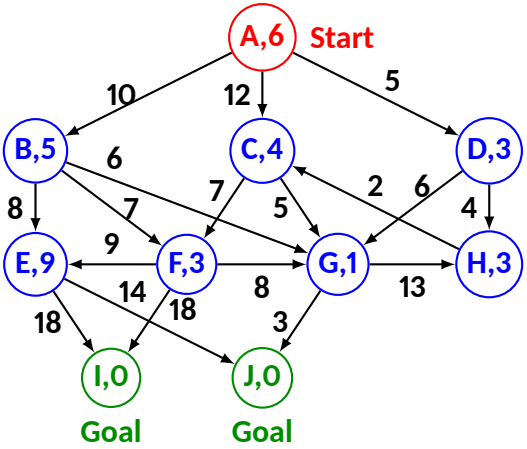
Cutoff OPEN

Execution of IDA*



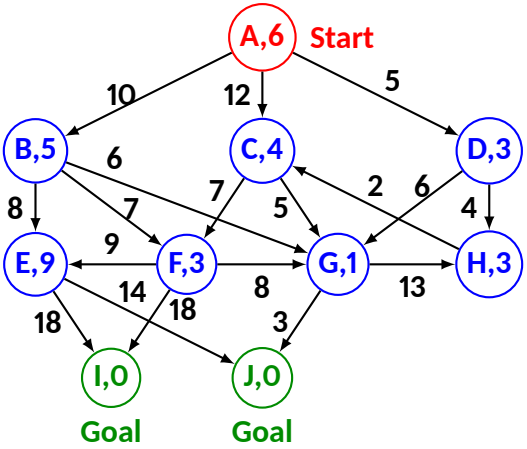
Cutoff	OPEN
6	(B,15), (C,16), (D,8)

Execution of IDA*



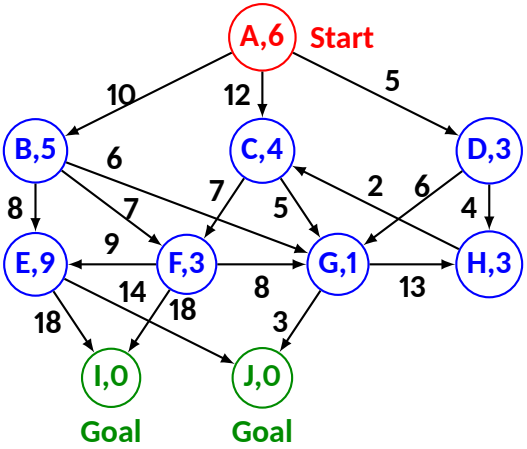
Cutoff	OPEN
6	(B,15), (C,16), (D,8)
8	(B,15), (C,16), (G,12), (H,12)

Execution of IDA*



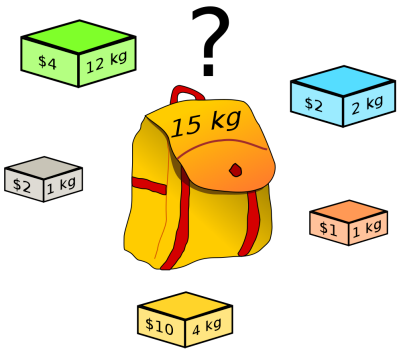
Cutoff	OPEN
6	(B,15), (C,16), (D,8)
8	(B,15), (C,16), (G,12), (H,12)
12	(B,15), (C,15), (J,14)

Execution of IDA*



Cutoff	OPEN
6	(B,15), (C,16), (D,8)
8	(B,15), (C,16), (G,12), (H,12)
12	(B,15), (C,15), (J,14)
14	Terminate

Maximization problem



- $h(n)$ should be an overestimate of the actual cost
- In the A* algorithm, choose the node that has the largest $f(n)$ value for expansion
- What will be the state-space?
- Develop some heuristic functions and apply A*, IDA*

image source:Wikipedia

Thank you!