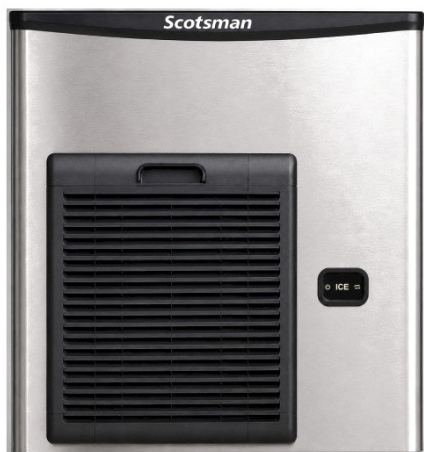


Łuskarka do lodu chłodzona wodą 280 kg / 24h, łuski lodu o niskiej zawartości wody | Scotsman MXF 627 WS



Krótki opis produktu:

Łuskarka Scotsman MXF 627 WS,
wydajność 280 kg/24 h, chłodzenie wodą.
Rodzaj lodu: łuski lodu o niskiej zawartości
wody.

Kody produktu

Reference: MXF627WS

EAN13: -

UPC: -

Cena produktu:

34 312,08 zł z
podatkiem

Opis produktu:

Downloaded at: 11:53 11 September 2009

1. **Definition:** A **graph** is a mathematical structure consisting of a set of vertices (nodes) and a set of edges (connections) between them. It is denoted as $G = (V, E)$, where V is the set of vertices and E is the set of edges.

2. **Types of Graphs:**

- Undirected Graph:** Edges do not have a direction. If there is an edge between vertices u and v , it can be traversed in both directions.
- Directed Graph (Digraph):** Edges have a direction. If there is an edge from vertex u to vertex v , it can only be traversed from u to v .
- Weighted Graph:** Edges have associated numerical values (weights) representing distance, cost, or capacity.
- Unweighted Graph:** Edges do not have associated numerical values.
- Complete Graph:** Every vertex is connected to every other vertex in the graph.
- Star Graph:** A central vertex is connected to all other vertices, which are not connected to each other.
- Path Graph:** A sequence of vertices connected linearly, forming a single path.
- Tree Graph:** A connected graph with no cycles (closed loops).
- Hamiltonian Graph:** A graph that contains a Hamiltonian cycle, which visits every vertex exactly once and returns to the starting point.

3. **Graph Traversal Algorithms:**

- Breadth-First Search (BFS):** Explores all vertices at the current depth level before moving on to the next level. It finds the shortest path from a source vertex to all other vertices in an unweighted graph.
- Depth-First Search (DFS):** Explores as far as possible along each edge before backtracking. It is used for finding cycles, connected components, and topological sorting.
- Dijkstra's Algorithm:** Finds the shortest path from a source vertex to all other vertices in a weighted graph with non-negative edge weights.
- Prim's Algorithm:** Finds the Minimum Spanning Tree (MST) of a weighted, undirected graph.
- Kruskal's Algorithm:** Finds the Minimum Spanning Tree (MST) of a weighted, undirected graph.

4. **Applications of Graphs:**

- Network Routing:** Finding the shortest path between two nodes in a network (e.g., internet routing, road navigation).
- Social Networks:** Modeling relationships between individuals (vertices) and their connections (edges).
- Project Scheduling:** Representing tasks (vertices) and their dependencies (edges) to determine the critical path and project duration.
- Image Segmentation:** Modeling pixels (vertices) and their spatial relationships (edges) to identify distinct regions in an image.
- Recommendation Systems:** Modeling user preferences (vertices) and item characteristics (edges) to suggest relevant items.
- Clustering Analysis:** Identifying groups of similar data points (vertices) based on their relationships (edges).
- Game Theory:** Modeling players (vertices) and their possible moves/strategies (edges) to analyze optimal game outcomes.

5. **Graph Theory Concepts:**

- Degree of a Vertex:** The number of edges incident to a vertex.
- Path:** A sequence of vertices connected by edges, where no vertex is repeated.
- Cycle:** A closed path where the starting and ending vertices are the same, and no other vertex is repeated.
- Connected Graph:** A graph where there is a path between every pair of vertices.
- Disconnected Graph:** A graph that consists of two or more separate connected components.
- Spanning Tree:** A subgraph that is a tree and includes all vertices of the original graph.
- Minimum Spanning Tree (MST):** A spanning tree with the minimum total edge weight.
- Maximum Spanning Tree (MaxST):** A spanning tree with the maximum total edge weight.

6. **Graph Representation:**

- Adjacency List:** A collection of lists, where each list represents the neighbors of a vertex.
- Adjacency Matrix:** A square matrix where the entry at row i and column j represents the weight of the edge between vertices i and j .
- Incidence Matrix:** A matrix where the rows represent vertices and the columns represent edges, with entries indicating the incidence of a vertex with an edge.

7. **Graph Complexity:**

- Time Complexity:** The amount of time an algorithm takes to run, often expressed in terms of the number of vertices (V) and edges (E).
- Space Complexity:** The amount of memory an algorithm uses, often expressed in terms of the number of vertices (V) and edges (E).

8. **Graph Theory Problems:**

- Shortest Path Problem:** Finding the path with the minimum total weight between two vertices.
- Traveling Salesman Problem (TSP):** Finding the shortest possible route that visits every vertex exactly once and returns to the starting point.
- Graph Coloring Problem:** Assigning colors to vertices such that no two adjacent vertices share the same color.
- Graph Isomorphism Problem:** Determining if two graphs are structurally identical (isomorphic).
- Graph Automorphism Problem:** Finding symmetries within a graph.

9. **Graph Theory in Real-World Applications:**

- Network Design:** Optimizing the layout of communication networks, power grids, and transportation systems.
- Biological Networks:** Modeling interactions between genes, proteins, and cells in a biological system.
- Financial Networks:** Analyzing relationships between financial institutions and markets.
- Web Graphs:** Studying the structure and dynamics of the World Wide Web.
- Supply Chain Management:** Optimizing the flow of goods and information between suppliers, manufacturers, and distributors.
- Urban Planning:** Designing efficient road networks and public transportation systems.
- Marketing and Advertising:** Identifying influential individuals and communities for targeted marketing campaigns.
- Security and Cybersecurity:** Analyzing network vulnerabilities and detecting malicious activity.

10. **Graph Theory Research Frontiers:**

- Complex Networks:** Studying the properties and dynamics of large, interconnected networks with non-trivial structures.
- Dynamic Graphs:** Analyzing graphs that change over time, where vertices and edges can be added or removed.
- Graph Machine Learning:** Applying machine learning techniques to analyze and predict properties of graphs.
- Quantum Graphs:** Exploring the intersection of graph theory and quantum mechanics.
- Topological Graphs:** Studying the properties of graphs that are invariant under continuous deformations.

- **Wydajność (kg):** 280
- **Rodzaj lodu:** lód o niskiej zawartości wody
- **Chłodzenie:** Woda
- **Czynnik chłodniczy:** R290
- **Zasilanie:** elektryczne
- **Napięcie zasilania:** 230 V
- **Szerokość (mm):** 625
- **Głębokość (mm):** 775

-
- Wysokość (mm): 690

Cechy produktu:

Gwarancja: 3 lata*

Czas realizacji: 1-21 dni

Wydajność (kg): 280

Szerokość (mm): 625

Głębokość (mm): 775

Wysokość (mm): 690

Linia: MXF

Model: MXF 627 WS

Rodzaj lodu: łuski lodu o niskiej zawartości wody

Chłodzenie: Woda

Czynnik chłodniczy: R290

Zasilanie: elektryczne

Napięcie zasilania: 230 V

Atrybuty produktu:

Galeria produktu:

