**8-тажриба иши**

**ТРАНСПОРТ МАСАЛАСИНИ ЧИЗИҚЛИ ПРОГРАММАЛАШ УСУЛЛАРИ БИЛАН ЕЧИШ**

**ТРАНСПОРТ МАСАЛАСИНИ ДАСТЛАБКИ ЕЧИМИНИ ШИМОЛИЙ ҒАРБ БУРЧАК УСУЛИДА АНИҚЛАШ**

В хозяйстве за время уборки при заготовке силоса необходимо перевезти 4000 т зеленой массы с пяти полей к четырем фермам, в том числе с первого поля 600 т, со второго – 240 т, третьего – 1360 т, четвертого – 1000 т и пятого – 900 т. Для первой фермы требуется 600т зеленой массы, второй – 800 т, третьей – 1400 т и четвертой – 1200 т.

Расстояние перевозки зеленой массы с полей к фермам приведено в таблице.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Поле | Объем заготовки силоса  (Поставщик) | Ферма | | | |
| 1-я | 2-я | 3-я | 4-я |
| 1-е | 600 | 5 | 6 | 2 | 2 |
| 2-е | 240 | 9 | 7 | 4 | 6 |
| 3-е | 1360 | 7 | 1 | 4 | 5 |
| 4-е | 1000 | 5 | 2 | 2 | 4 |
| 5-е | 900 | 6 | 4 | 3 | 4 |
| Потребитель |  | 600 | 800 | 1400 | 1200 |

Требуется составить такой план перевозок, чтобы общие транспортные затраты были минимальными.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Transport Masalasi</title>

    <link rel="icon" href="cotton.jpg" type="image/icon type">

    <script src="https://cdn.tailwindcss.com"></script>

    <script>

        function generateTable() {

            let suppliers = parseInt(document.getElementById("suppliers").value);

            let consumers = parseInt(document.getElementById("consumers").value);

            let table = document.getElementById("dynamicTable");

            table.innerHTML = "";

            let headerRow = table.insertRow();

            headerRow.insertCell().outerHTML = "<th class='border bg-gray-200 px-6 py-3 text-left'>Dalalar</th>";

            headerRow.insertCell().outerHTML = "<th class='border bg-gray-200 px-6 py-3 text-left'>Resurs</th>";

            for (let j = 0; j < consumers; j++) {

                headerRow.insertCell().outerHTML = `<th class='border bg-gray-200 px-6 py-3 text-left'>F${j + 1}</th>`;

            }

            for (let i = 0; i < suppliers; i++) {

                let row = table.insertRow();

                row.classList.add(i % 2 === 0 ? 'bg-gray-50' : 'bg-white');

                row.insertCell().innerHTML = `<td class='border px-6 py-3'>A${i + 1}</td>`;

                let supplyCell = row.insertCell();

                supplyCell.innerHTML = `<input type='number' class='w-20 p-2 border rounded'/>`;

                for (let j = 0; j < consumers; j++) {

                    let cell = row.insertCell();

                    cell.innerHTML = `<input type='number' class='w-20 p-2 border rounded'/>`;

                }

            }

            let demandRow = table.insertRow();

            demandRow.classList.add('bg-gray-200');

            demandRow.insertCell().outerHTML = "<th class='border px-6 py-3'>Talab</th>";

            demandRow.insertCell();

            for (let j = 0; j < consumers; j++) {

                let demandCell = demandRow.insertCell();

                demandCell.innerHTML = `<input type='number' class='w-20 p-2 border rounded'/>`;

            }

        }

        function getTableData() {

            let table = document.getElementById("dynamicTable");

            let rows = table.rows.length - 1;

            let cols = table.rows[0].cells.length - 2;

            let costs = [];

            let supply = [];

            let demand = [];

            for (let i = 1; i < rows; i++) {

                let rowData = [];

                supply.push(parseFloat(table.rows[i].cells[1].querySelector("input").value) || 0);

                for (let j = 2; j < cols + 2; j++) {

                    rowData.push(parseFloat(table.rows[i].cells[j].querySelector("input").value) || 0);

                }

                costs.push(rowData);

            }

            for (let j = 2; j < cols + 2; j++) {

                demand.push(parseFloat(table.rows[rows].cells[j].querySelector("input").value) || 0);

            }

            return { costs, supply, demand };

        }

        function vogelApproximation(costs, supply, demand) {

            let rows = costs.length;

            let cols = costs[0].length;

            let allocation = Array.from({ length: rows }, () => Array(cols).fill(0));

            let steps = [];

            while (supply.some(s => s > 0) && demand.some(d => d > 0)) {

                let minCost = Infinity, minI = -1, minJ = -1;

                for (let i = 0; i < rows; i++) {

                    for (let j = 0; j < cols; j++) {

                        if (costs[i][j] < minCost && supply[i] > 0 && demand[j] > 0) {

                            minCost = costs[i][j];

                            minI = i;

                            minJ = j;

                        }

                    }

                }

                let quantity = Math.min(supply[minI], demand[minJ]);

                allocation[minI][minJ] = quantity;

                steps.push(`A${minI + 1} -> F${minJ + 1}: ${quantity} birlik`);

                supply[minI] -= quantity;

                demand[minJ] -= quantity;

            }

            return { allocation, steps };

        }

        function calculateTotalCost() {

            let { costs, supply, demand } = getTableData();

            let { allocation, steps } = vogelApproximation(costs, supply, demand);

            let totalCost = 0;

            let explanation = "<strong>Hisoblash jarayoni:</strong><br>";

            for (let i = 0; i < costs.length; i++) {

                for (let j = 0; j < costs[i].length; j++) {

                    totalCost += allocation[i][j] \* costs[i][j];

                }

            }

            explanation += steps.join("<br>");

            document.getElementById("result").innerHTML = `Minimal transport xarajati: <strong>${totalCost}</strong><br><br>${explanation}`;

        }

    </script>

</head>

<body class="bg-gray-100 p-10">

    <div class="max-w-4xl mx-auto bg-white p-6 shadow rounded-lg">

        <h2 class="text-2xl font-bold mb-4">Transport Masalasi</h2>

        <div class="mb-4">

            <label for="suppliers" class="mr-2">Dalalar soni:</label>

            <input type="number" id="suppliers" min="1" value="5" class="border p-2 rounded w-20">

            <label for="consumers" class="ml-4 mr-2">Ferma soni:</label>

            <input type="number" id="consumers" min="1" value="4" class="border p-2 rounded w-20">

            <button onclick="generateTable()" class="bg-blue-500 text-white px-4 py-2 rounded ml-4 hover:bg-blue-600">Jadval yaratish</button>

        </div>

        <table class="table-auto border-collapse border w-full text-center shadow-md rounded-lg" id="dynamicTable"></table>

        <br>

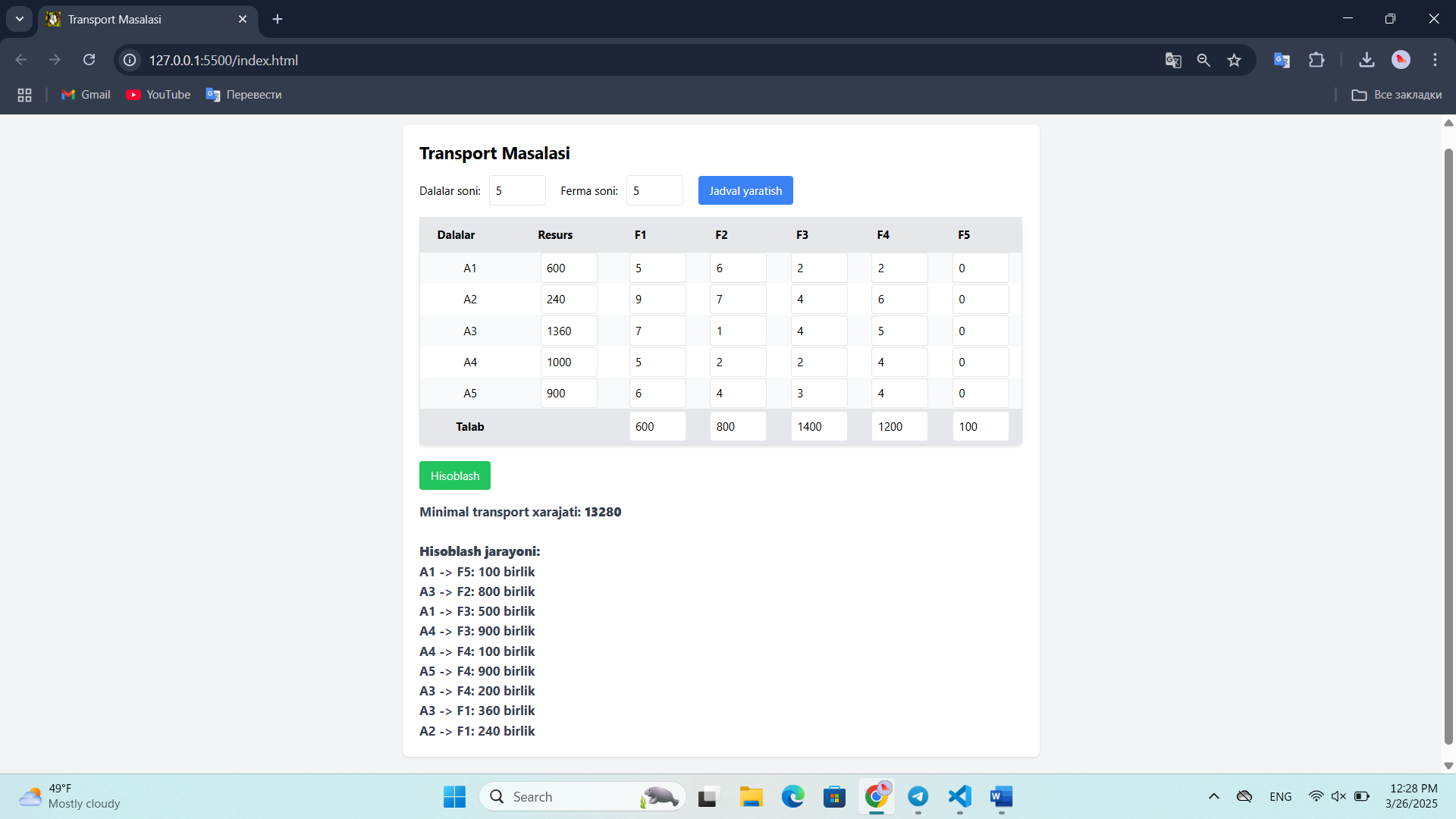
        <button onclick="calculateTotalCost()" class="bg-green-500 text-white px-4 py-2 rounded hover:bg-green-600">Hisoblash</button>

        <p id="result" class="mt-4 font-bold text-lg text-gray-700"></p>

    </div>

</body>

</html>



Optimal xarajat: 13280