ГУАП

КАФЕДРА № 12

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доц., канд. техн. наук |  |  |  | В.Е. Таратун |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ № 2 |
| РАЗРАБОТКА ПРИЛОЖЕНИЯ ПО РАСПРЕДЕЛЕНИЮ РЕСУРСОВ В СИСТЕМЕ НА ОСНОВЕ МЕТОДА СЕВЕРО-ЗАПАДНОГО УГЛА |
| по курсу: ПРОГРАММИРОВАНИЕ НА ЯЗЫКАХ ВЫСОКОГО УРОВНЯ |
|  |
|  |

РАБОТУ ВЫПОЛНИЛА

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 1921 |  |  |  | А.В. Музыченко |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2020

1. **Цель работы и задачи**

**Цели:**

* Выработать практические навыки работы с системой Visual Studio на основе языка программирования высокого уровня – С++;
* Изучить метод построения опорного плана, используя метод северо-западного угла;
* Научиться создавать, вводить в компьютер, выполнять и исправлять простейшие программы.

Задачи:

* Разработать алгоритм в виде блок-схемы для распределения ресурсов между элементами, используя метод северо-западного угла;
* Разработать консольную программу для получения начального опорного плана распределения ресурсов, используя метод северо- западного угла.

1. **Исходные данные**

Условие задачи:

У поставщика одежды имеется три склада, ему нужно предоставить свой товар четырем потребителям (двум индивидуальным и двум посредникам). Используя метод северо – западного угла наиболее эффективно распределить ресурсы со склада и доставить их потребителям. (стоимость перераспределения известна и записана в таблице ниже).

Представим решение задачи в виде матрицы:

Метод с-з угла: на первом складе 7 шт товара, первому потребителю нужно 2 шт товара, отправляем две на складе остается 5. Следующему потребителю нужно 4 шт товара, отправляем с первого склада, на нем остается 1 шт товара, отправляем ее третьему потребителю, ему еще нужно 5 шт товара, отправляем их со второго склада, и четвертому потребителю с третьего склада 10 шт товара. Задача выполнена, товары доставлены.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | | | | | Запасы |
| 1 | 2 | | 3 | | 4 | |
| 1 | 20 | | 30 | | 50 | | 20 | 7 |
| 2 | 70 | | 30 | | 40 | | 70 | 5 |
| 3 | 40 | | 10 | | 50 | | 20 | 10 |
| Потребности | 2 | | 4 | | 6 | | 10 | 22 |

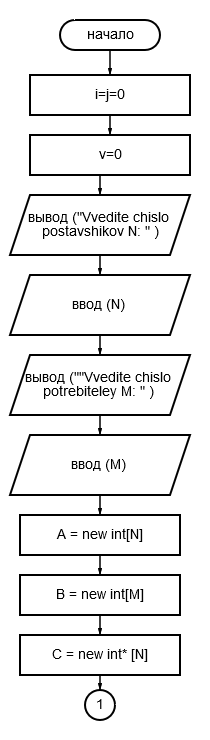
Элементы, стоящие в клетках матрицы, соответствуют затратам, отвечающим выделению одной единицы ресурса на работу. То есть это заданная стоимость перераспределения.

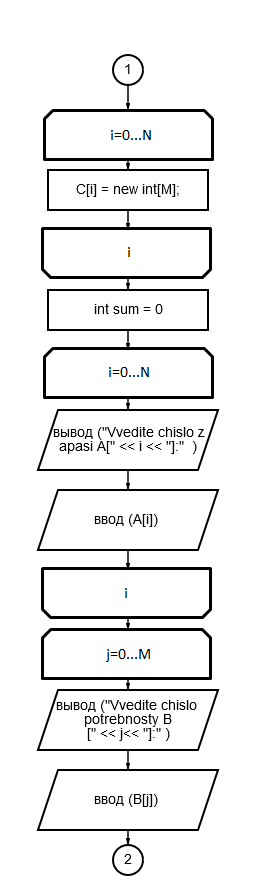
Результат метода с-з угла

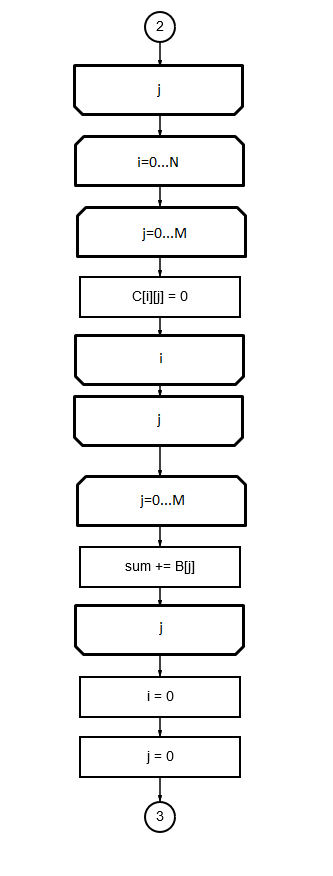
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Число потребителей | | | | Запасы (сколько товаров на складах) | остатки | |
| Число  постав-  щиков | 2(20) | 4(30) | 1(50) | 0 | 7 | 5 | 1 |
| 0 | 0 | 5(40) | 0 | 5 |  |  |
| 0 | 0 | 0 | 10(20) | 10 |  |  |
|  | 2 | 4 | 6 | 10 | 22 |  |  |
| Число потребностей (сколько нужно товара потребителям) | | | |  |  |  |

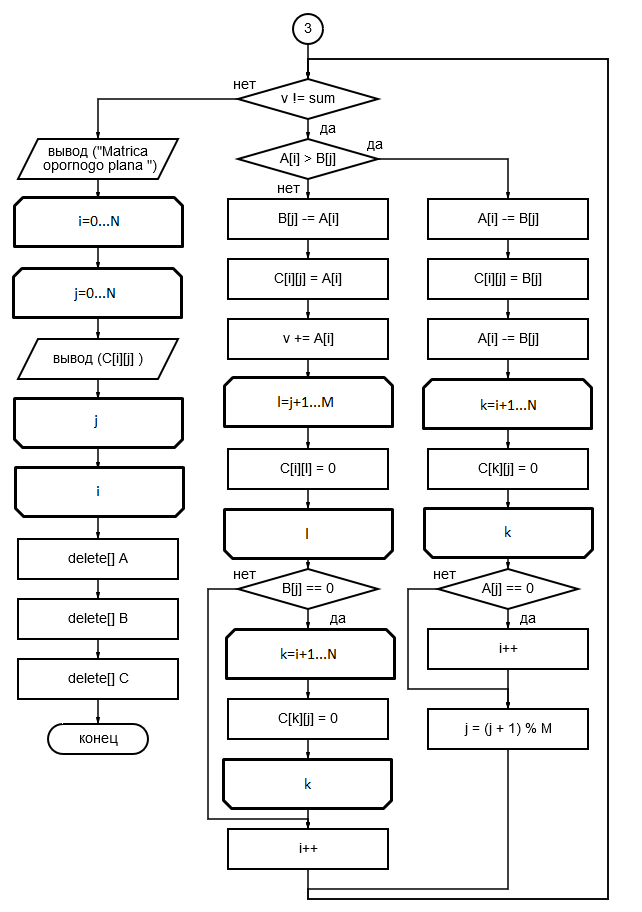
Подсчитаем затраты, соответствующие этому решению. Они равны сумме произведений количества ресурсов на стоимость перераспределения одной единицы ресурса.

1. **Алгоритм в виде блок-схемы**









1. **Листинг программного кода**

#include <iostream>

using namespace std;

void main()

{

int i, j;

i = j = 0;

int\* A;

int\* B;

int\*\* C;

int v = 0;

int N, M;

std::cout << "Vvedite chislo postavshikov N: " << std::endl;

std::cin >> N;

std::cout << "Vvedite chislo potrebiteley M: " << std::endl;

std::cin >> M;

A = new int[N];

B = new int[M];

C = new int\* [N];

for (i = 0; i < N; i++)

{

C[i] = new int[M];

}

int sum = 0;

for (i = 0; i < N; i++)

{

std::cout << "Vvedite chislo zapasi A[" << i << "]:" << std::endl;

std::cin >> A[i];

}

for (j = 0; j < M; j++)

{

std::cout << "Vvedite chislo potrebitely B[" << j<< "]:" << std::endl;

std::cin >> B[j];

}

for (i = 0; i < N; i++)

{

for (j = 0; j < M; j++)

{

C[i][j] = 0;

}

}

for (j = 0; j < M; j++)

{

sum += B[j];

}

i = 0;

j = 0;

while (v != sum)

{

if (A[i] > B[j])

{

A[i] -= B[j];

C[i][j] = B[j];

v += B[j];

for (int k = i + 1; k < N; k++)

{

C[k][j] = 0;

}

if (A[j] == 0)

{

i++;

}

j = (j + 1) % M;

}

else

{

B[j] -= A[i];

C[i][j] = A[i];

v += A[i];

for (int l = j + 1; l < M; l++)

{

C[i][l] = 0;

}

if (B[j] == 0)

{

for (int k = i + 1; k < N; k++)

{

C[k][j] = 0;

}

}

i++;

}

}

std::cout << "Matrica opornogo plana \n";

for (i = 0; i < N; i++)

{

for (j = 0; j < M; j++)

{

std::cout << C[i][j] << " ";

}

std::cout << std::endl;

}

delete[] A;

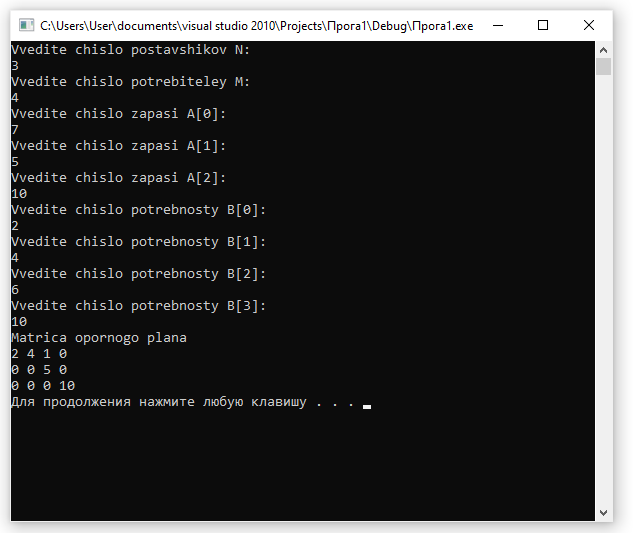
delete[] B;

delete[] C;

system("pause");

}

1. **Результат работы программы**



1. **Вывод**

Разработала алгоритм в виде блок-схемы для распределения ресурсов между элементами, используя метод северо-западного угла, а также консольную программу для получения начального опорного плана распределения ресурсов. Посчитали затраты, для поставленной задачи (610 ед. стоимости).