1. **Užduotis.** **Pirkėjų srautai**

Darbo laikas skirstomas taip: 8.00-8.30 – 1 pusvalandis; 8.30-9.00 – 2 pusvalandis, ..., 19.30-20.00 – 24 pusvalandis.

Pradinių duomenų failo **pirkejai\_data.txt** pirmoje eilutėje įrašytas per dieną aptarnautų pirkėjų skaičius **n**. Tolesnėse **n** eilučių įrašyta po 2 sveikuosius skaičius – kasos čekyje nurodytas pirkėjo aptarnavimo laikas – valanda ir minutė.

Rezultatų faile **pirkejai\_res.txt** turi būti išvardinti visi 24 pusvalandžiai, nurodant pusvalandžio pradžios valandą ir minutę, pabaigos valandą ir minutę, bei tą pusvalandį aptarnautų pirkėjų skaičių. Vienam pusvalandžiui skiriama viena eilutė.

Jei pirkėjo aptarnavimo laikas sutampa su vieno pusvalandžio pabaiga, o kito pradžia, tokį pirkėją priskirkite ankstesniam pusvalandžiui (tam pusvalandžiui, su kurio pabaiga sutampa pirkėjo aptarnavimo laikas).

|  |  |
| --- | --- |
| pirkejai\_data.txt | pirkejai\_data.txt |
| 5  8 45  17 25  18 45  19 13  19 35 | 8 0 8 30 0  8 30 9 0 1  9 0 9 30 0  9 30 10 0 0  10 0 10 30 0  10 30 11 0 0  11 0 11 30 0  11 30 12 0 0  12 0 12 30 0  12 30 13 0 0  13 0 13 30 0  13 30 14 0 0  14 0 14 30 0  14 30 15 0 0  15 0 15 30 0  15 30 16 0 0  16 0 16 30 0  16 30 17 0 0  17 0 17 30 1  17 30 18 0 0  18 0 18 30 0  18 30 19 0 1  19 0 19 30 1  19 30 20 0 1 |

1. **Reikalavimai**

* Main funkcijoje privalo būti tik lokaliųjų kintamųjų apsirašymas bei skaitymo ir rašymo funkcijos.
* Sukurti funkciją **Skaitymas**.
* Sukurti funkciją **Rasymas**.
* Nenaudoti globalių kintamųjų.

1. **Programos tekstas C++ kalba**

#include <iostream>

#include <iomanip>

#include <cmath>

#include <fstream>

using namespace std;

void Skaito (int A[], int &n);

void Raso(int A[], int n);

void Laikas(int A[], int n, int B[]);

void Klientura(int A[], int n, int val, int minut, int minut1, int &klient);

int main(){

int A[1000], n, B[1000];

Skaito(A, n);

Laikas(A, n, B);

Raso(B, 120);

return 0;

}

void Skaito (int A[], int &n)

{

ifstream df ("pirkejai\_data.txt");

df >> n;

for (int i = 0; i < n \* 2; i++)df >> A[i];

df.close();

}

void Raso(int A[], int n)

{

ofstream rf ("pirkejai\_res.txt");

for (int i = 0; i < n; i = i + 5)rf << A[i] << " " << A[i + 1] << " " << A[i + 2] << " " << A[i + 3] << " " << A[i + 4] << endl;

rf.close();

}

void Laikas(int A[], int n, int B[])

{

int ind = 0, k = 0, i1 = 0;

for (int i = 8; i < 25; i = i + 1){

for(int i1 = 0; i1 <= 30; i1 = i1 + 30){

B[ind] = i;

ind++;

B[ind] = i1;

ind++;

}

Klientura(A, n, i, 0, 30, k);

B[ind] = k;

ind++;

for(int i1 = 30; i1 >= 0; i1 = i1 - 30){

B[ind] = i;

ind++;

B[ind] = i1;

ind++;

}

k = 0;

B[ind - 2] = B[ind - 2] + 1;

Klientura(A, n, i, 30, 60, k);

B[ind] = k;

k = 0;

ind++;

}

}

void Klientura(int A[], int n, int val, int minut, int minut1, int &klient)

{

for(int i = 0; i < n \* 2; i = i + 2){

if (A[i] == val && minut < A[i + 1] && A[i + 1] < minut1)klient++;

else if (minut1 == 60 && A[i] == val + 1 && minut1 % 60 == A[i + 1])klient++;

else if (minut1 == 30 && A[i] == val && minut1== A[i + 1])klient++;

}

}