**A elaborat: Jiganov Andrei, gr. IS11Z**

**A verificat: Vitalie Ticau, lector universitar**

**PROIECT DE PROGRAME NR. 1. ALGORITMI CLASICI**

**TEMA: NUMERE PALINDROAME**

**I. Formularea problemei**

1. De verificat dacă numărul *nr* este palindrom.
2. De indicat toate numerele din intervalul *[a; b]*, care satisfac proprietatea de bază.
3. De indicat primele *n* numere, care satisfac proprietatea de bază.
4. De indicat toate numerele din intervalul *[a; b]*, care satisfac proprietatea de bază (varianta optimizată) sau care satisfac a doua proprietate.
5. De determinat cel mai mare număr, care satisface condiţia indicată.

**II. Studiul preliminar**

Un palindrom este un șir de caractere (de obicei cuvinte, fraze sau numere) care citit de la stânga la dreapta sau de la dreapta la stânga rămâne neschimbat. **Ex.:** 11, 22, 33, 44, 55, 66, 77, 88, 99, 101, 111, 121, 131, etc.

**III. Programul**

#include <iostream>

#include <cmath>

#include <iomanip>

using namespace std;

int Palindrom(int);

int main()

{

cout << "Proiect ne. 1, elaborat de Andrei Jiganov" << endl;

cout << "Tema: Palindrom" << endl;

// 1. Проверить если число является Палиндромом.

int numarMin, numarMax, barrier;

// 2. Определить все палиндромы заданного интервала.

int sum = 0;

int aux = 0;

cout << "Introduceti de la ce numar si pina la ce numar o sa prelucreze programmul:";

cin >> numarMin >> numarMax;

// 3. Определить минимальный и максимальный палиндром заданного интервала.

cout << "Introduceti cite numere vreti sa afisati:" << endl;

int Min = numarMax;

int Max = 0;

cout << "Numerele Polindroame:" << endl;

for (int i = numarMin; i < numarMax; i++)

{

if (i >= 11)

{

if (Palindrom(i))

{

if (Max < i)

{

Max = i;

}

if (Min > i)

{

Min = i;

}

// 4. Определение количества полиндромов в заданном интервале.

sum++;

cout << i << endl;

}

}

}

cin >> barrier;

// 5. Вывести первые n палиндромов

for (numarMin; numarMin < numarMax; numarMin++)

{

if (numarMin >= 10)

{

if (Palindrom(numarMin))

{

if (aux < barrier)

{

cout << "\n" << numarMin << " - Numar din intervalul dat" << endl;

aux++;

}

}

}

}

cout << "\n" << "Min = " << Min << endl; // Минимум

cout << "Max = " << Max << endl; // Максимум

cout << "\n" << "Total: " << sum; // Общее кол-во палиндромов в заданном интервале

return 0;

}

int Palindrom(int n)

{

int invers = 0;

int aux = n;

while (aux > 0)

{

invers = invers \* 10 + aux % 10;

aux /= 10;

}

return n == invers;

}

**IV. Rezultatele**

Numerele Palindroame din intervalul de 0 pana la 1000:

11 22 33 44 55 66 77 88 99 101 111 121 131 141 151 161 171 181 191 202 212 222 232 242 252 262 272 282 292 303 313 323 333 343 353 363 373 383 393 404 414 424 434 444 454 464 474 484 494 505 515 525 535 545 555 565 575 585 595 606 616 626 636 646 656 666 676 686 696 707 717 727 737 747 757 767 777 787 797 808 818 828 838 848 858 868 878 888 898 909 919 929 939 949 959 969 979 989 999

Primele 10:

11 - Numar din intervalul dat

22 - Numar din intervalul dat

33 - Numar din intervalul dat

44 - Numar din intervalul dat

55 - Numar din intervalul dat

66 - Numar din intervalul dat

77 - Numar din intervalul dat

88 - Numar din intervalul dat

99 - Numar din intervalul dat

101 - Numar din intervalul dat

Min = 11

Max = 999

Total: 99

**V. Concluzii**

1. Calcularea palindroamelor se incepe de la 11, este cel mai mic palindrom.