

Міністерство освіти і науки України

**Національний технічний університет України «Київський
політехнічний інститут імені Ігоря Сікорського»**

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 5 з дисципліни

«Основи програмування

1. Базові конструкції»

«Організація циклічних процесів. Складні цикли»

Варіант 22

Виконав студент ІП-14 Нікулін Павло Юрійович
(шифр, прізвище, ім'я, по батькові)

Перевірів Новікова Поліна Анатоліївна
(прізвище, ім'я, по батькові)

Лабораторна робота №5

Організація циклічних процесів. Складні цикли

Мета: вивчити особливості організації складних циклів.

Хід роботи

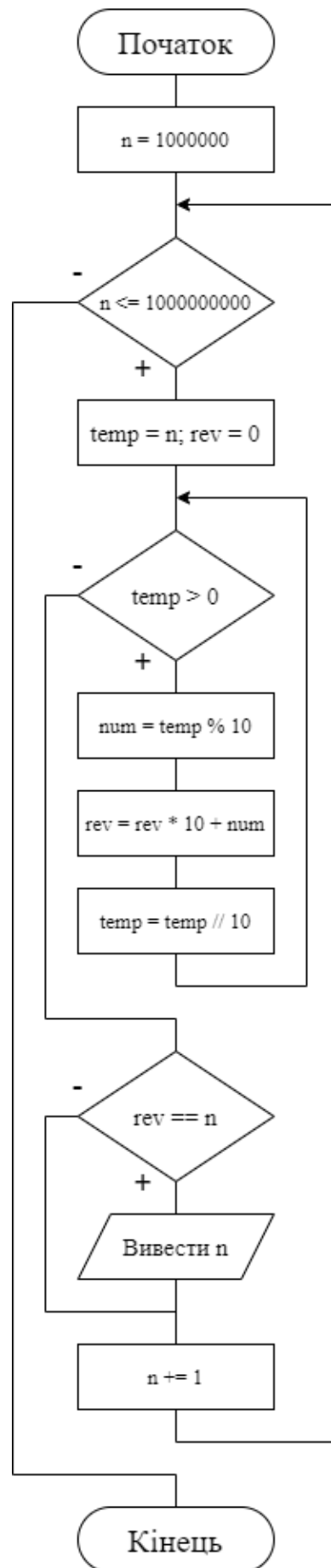
Задача: натуральне число називається паліндромом, якщо його запис читається однаково з початку та з кінця. Знайти всі паліндроми з інтервалу [1 000 000, 1 000 000 000].

Розв'язання

1. **Постановка задачі:** результатом роботи має бути всі натуральні числа, що є паліндромами на заданому проміжку. Для пошуку паліндромів потрібно «перегорнути» число n . Якщо воно буде дорівнювати числу n , то це паліндром.
2. Побудова **математичної моделі**. Складемо таблицю імен змінних.

| Змінна | Тип | Ім'я | Призначення |
|---------------------------|------------|------|----------------|
| Число з проміжку | Натуральне | N | Початкове дане |
| Число для операцій | Натуральне | TEMP | Проміжне дане |
| Цифра числа з проміжку | Натуральне | NUM | Проміжне дане |
| Реверсія числа з проміжку | Натуральне | REV | Проміжне дане |

Блок-схема



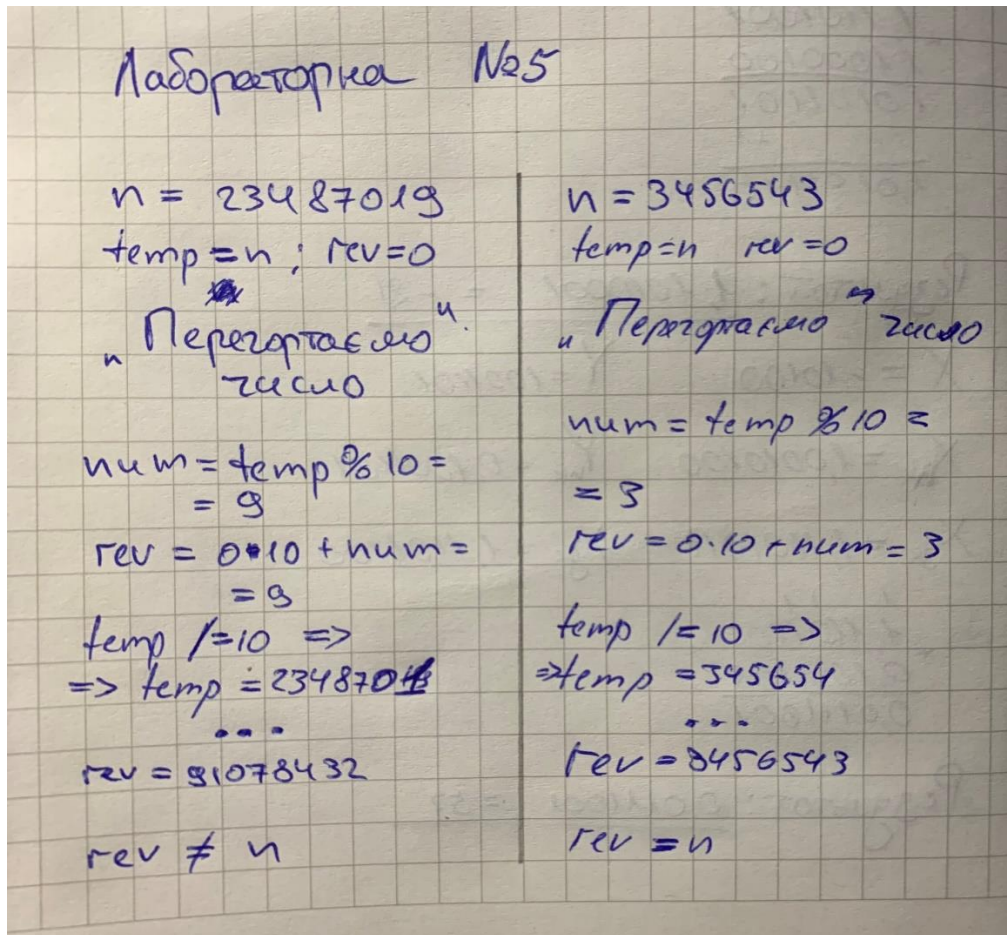
C++

Код

```
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      int n = 1000000, x = 0;
7
8      /*Задаємо інтервал числа x*/
9      while (n <= 1000000000)
10     {
11         int temp = n, rev = 0, num;
12
13         /*Перегортаємо" числа*/
14         while (temp > 0)
15         {
16             num = temp % 10;
17             rev = (rev * 10) + num;
18             temp = temp / 10;
19         }
20
21         /*Порівнюємо число x з його дзеркальним відображенням*/
22         if (rev == n)
23         {
24             cout << n << " ";
25             x++;
26         }
27
28         n += 1;
29     }
30
31     /*Виводимо число паліндром на інтервалі*/
32     cout << "\n" << x << " palindroms";
33 }
```

Результат (весь результат не вміщається на скріншоті)

```
99 995040599 995050599 995060599 995070599 995080599 995090599 995101599 995111599 995121599 995131599 995141599 995151599 995161599 995171599 995181599 995191599 995202599 995212599 995222599 995232599 995242599
99 995252599 995262599 995272599 995282599 995292599 995303599 995313599 995323599 995333599 995343599 995353599 995363599 995373599 995383599 995393599 995404599 995414599 995424599 995434599 995444599 995454599
99 995464599 995474599 995484599 995494599 995505599 995515599 995525599 995535599 995545599 995555599 995565599 995575599 995585599 995595599 995606599 995616599 995626599 995636599 995646599 995656599 995666599
995676599 995686599 995696599 995707599 995717599 995727599 995737599 995747599 995757599 995767599 995777599 995787599 995797599 995808599 995818599 995828599 995838599 995848599 995858599 995868599 995878599 99
995888599 995898599 995909599 995919599 995929599 995939599 995949599 995959599 995969599 995979599 995989599 995999599 996000699 996010699 996020699 996030699 996040699 996050699 996060699 996070699 996080699 99
996090699 996101699 996111699 996121699 996131699 996141699 996151699 996161699 996171699 996181699 996191699 996202699 996212699 996222699 996232699 996242699 996252699 996262699 996272699 996282699 996292699 996
996303699 996313699 996323699 996333699 996343699 996353699 996363699 996373699 996383699 996393699 996404699 996414699 996424699 996434699 996444699 996454699 996464699 996474699 996484699 996494699 996505699 9965
15699 996525699 996535699 996545699 996555699 996565699 996575699 996585699 996595699 996606699 996616699 996626699 996636699 996646699 996656699 996666699 996676699 996686699 996696699 996707699 996717699 99672
7699 996737699 996747699 996757699 996767699 996777699 996787699 996797699 996808699 996818699 996828699 996838699 996848699 996858699 996868699 996878699 996888699 996898699 996909699 996919699 996929699 996939
699 996949699 996959699 996969699 996979699 996989699 996999699 997000799 997010799 997020799 997030799 997040799 997050799 997060799 997070799 997080799 997090799 997101799 997111799 997121799 997131799 9971417
99 997151799 997161799 997171799 997181799 997191799 997202799 997212799 997222799 997232799 997242799 997252799 997262799 997272799 997282799 997292799 997303799 997313799 997323799 997333799 997343799 99735379
9 997363799 997373799 997383799 997393799 997404799 997414799 997424799 997434799 997444799 997454799 997464799 997474799 997484799 997494799 997505799 997515799 997525799 997535799 997545799 997555799 997565799
997575799 997585799 997595799 997606799 997616799 997626799 997636799 997646799 997656799 997666799 997676799 997686799 997696799 997707799 997717799 997727799 997737799 997747799 997757799 997767799 997777799
997787799 997797799 997808799 997818799 997828799 997838799 997848799 997858799 997868799 997878799 997888799 997898799 997909799 997919799 997929799 997939799 997949799 997959799 997969799 997979799 997989799 99
99799799 99800899 99801899 99802899 99803899 99804899 99805899 99806899 99807899 99808899 99809899 99810899 99811899 99812899 99813899 99814899 99815899 99816899 99817899 99818899 99819899 99820899 99821899 99822899
99823899 99824899 99825899 99826899 99827899 99828899 99829899 99830899 99831899 99832899 99833899 99834899 99835899 99836899 99837899 99838899 99839899 99840899 99841899 99842899 99843899 99844899 99845899 99846899
99847899 99848899 99849899 99850899 99851899 99852899 99853899 99854899 99855899 99856899 99857899 99858899 99859899 99860899 99861899 99862899 99863899 99864899 99865899 99866899 99867899 99868899 99869899 99870899 99871899
99872899 99873899 99874899 99875899 99876899 99877899 99878899 99879899 99880899 99881899 99882899 99883899 99884899 99885899 99886899 99887899 99888899 99889899 99890899 99891899 99892899 99893899 99894899 99895899 99896899
99897899 99898899 99899899 99900899 99901899 99902899 99903899 99904899 99905899 99906899 99907899 99908899 99909899 99910899 99911899 99912899 99913899 99914899 99915899 99916899 99917899 99918899 99919899 99920899 99921899 99922899
99923899 99924899 99925899 99926899 99927899 99928899 99929899 99930899 99931899 99932899 99933899 99934899 99935899 99936899 99937899 99938899 99939899 99940899 99941899 99942899 99943899 99944899 99945899 99946899 99947899
99948899 99949899 99950899 99951899 99952899 99953899 99954899 99955899 99956899 99957899 99958899 99959899 99960899 99961899 99962899 99963899 99964899 99965899 99966899 99967899 99968899 99969899 99970899 99971899 99972899
99973899 99974899 99975899 99976899 99977899 99978899 99979899 99980899 99981899 99982899 99983899 99984899 99985899 99986899 99987899 99988899 99989899 99990899 99991899 99992899 99993899 99994899 99995899 99996899 99997899
99998899 99999899 1000000000
100000 palindroms
```



Висновок

Під час виконання лабораторної роботи було досліджено особливості організації складних циклів. Було використано два цикли *while*: для задання інтервалу числа x та для «перегортання» цього ж числа. Також використано умовний оператор *if* для порівняння числа x та його дзеркального відображення. Використана структура вкладеного циклу з передумовою.