# Лабораторна Работа № 10

1. Множини
   1. Надбання навичок використання множин
      1. Оголошення множини. Звернення до елементу множини. Додавання елементу до множини. Вилучення елементу з множини. Складання множин. Віднімання множин. Перетин множин.

# ЗаВдання 1

Створіть програму, що має розв’язувати поставлену у вашому варіанті задачу. Програма повинна:

* являти собою консольний додаток;
* відображати відомості про призначення та можливості програми;
* відображати відомості про автора програми;
* містити уведення необхідних даних з клавіатури;
* виводити результати розрахунків на екран;
* код програми має мітити у собі необхідні пояснення з використанням ремарок;
* всі алгоритми повинні міститися у функціях або процедурах;
* код програми не має містити у собі позначки;
* програма має завершуватися запитом на повтор розрахунків.

# ЗаВдання 2

Створіть блок-схему алгоритму

Заданий рядок символів. Надрукувати множину приголосних букв без урахування регістру, що зустрічаються у рядку

**Програмний код:**

**uses** crt;

{Перелік приголосних букв}

**const**

symbols = ['Б', 'б', 'В', 'в', 'Г', 'г', 'Д', 'д', 'Ж', 'ж', 'З', 'з', 'Й', 'й', 'К', 'к', 'Л', 'л', 'М', 'м', 'Н', 'н', 'П', 'п', 'Р', 'р', 'С', 'с', 'Т', 'т', 'Ф', 'ф', 'Х', 'х', 'Ц', 'ц', 'Ч', 'ч', 'Ш', 'ш', 'Щ', 'щ'];

{Repeat Program}

**function** RepProg: boolean;

**var**

s: char;

**begin**

Write('Continue? Y/N: ');

readln(s);

**if** (s = 'Y') **or** (s = 'y') **or** (s = 'Н') **or** (s = 'н') **then**

RepProg := False

**else** RepProg := True;

**end**;

{Info About Program}

**procedure** InfoProg;

**begin**

TextColor(Black);

TextBackground(White);

clrscr;

WriteLn('1341.I1off');

WriteLn('Програма для виводу множини приголосних букв без урахування регiстру, що зустрiчаються у рядку');

WriteLn;

WriteLn;

**end**;

{Main Procedure}

**procedure** Main;

**var**

a: **array**[1..20]**of** string;

s, sl: string;

i, k, n, j: byte;

**begin**

Write('Введiть сиволи: ');

Readln(s);

s := s + ' ';

sl := '';

n := 0;

**for** i := 1 **to** length(s) **do**

sl := sl + s[i];

**if** length(sl) > 0 **then**

**begin**

k := 0;

**for** j := 1 **to** n **do**

**if** a[j] = sl **then**

k := k + 1;

**if** k = 0 **then**

**begin**

n := n + 1;a[n] := sl;

**end**;

sl := '';

**end**;

Write('Приголоснi: ');

**for** i := 1 **to** length(s) **do**

**if** s[i] **in** symbols **then** //Буква належить до приголосних

**begin**

s[i] := s[i].ToLower; //Прирівнюємо букву до нижнього регістру

write(s[i], ' ');// Виводимо букву на екран

**end**;

WriteLn;

**end**;

**begin**

**repeat**

InfoProg;

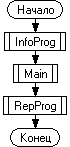
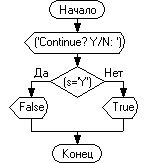
Main;

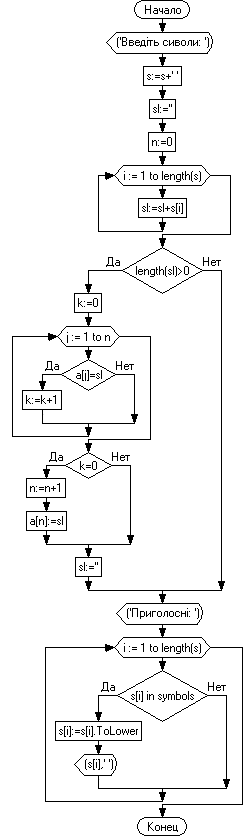
**until** RepProg;

**end**.

Блок-схема

Main body Function RepProg



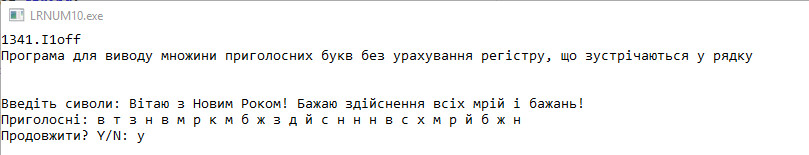
****

Рис 2. Результат програми

**Висновок** : На цій лабораторній роботі я за допомогою певних системних прийомів створив консольну програму застосувавши підключення модулів Pascal. Також робота була виконана за допомогою блок-оператора та с застосуванням визначених констант та змінних.

Основні функції,які були використані:

* Begin – початок роботи програми
* End – для завершення програми
* Var – вказівка функцій
* Const – вказівка констант
* If – умовний оператор
* Repeat – организація циклів з невідомим числом повторів, виконання циклу до тих пір доки не стане істинним деяка умова
* **For** – цикл, використовується, коли кількість повторень відома до початку циклу
* Lowercase – Повертає всі сиволи рядка в нижній регістр