**ЛАБОРАТОРНА РОБОТА № 24**

**Варіант 2**

Робота з рядками

***Мета*** : отримання практичних навиків роботи з рядками

**Хід роботи:**

**Завдання 1**: Ввести з клавіатури рядок символів і обробити його відповідно до свого варіанту, використовуючи функції.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <windows.h>

#include<stdio.h>

#include<math.h>

#include <conio.h>

#include <stdlib.h>

#include<string.h>

#include<ctype.h>

void zxc(char a[], int count) {

char\* sl;

for (int i = 0; i < strlen(a); i++) {

if (isupper(a[i]))

a[i] = tolower(a[i]);

}

sl = strtok(a, " .,-:");

while (sl != NULL) {

int t = 0;

for (int i = 0; i < strlen(sl); i++)

{

if (sl[i] == 'a' || sl[i] == 'e' || sl[i] == 'i' || sl[i] == 'o' || sl[i] == 'u' || sl[i] == 'y')

t++;

}

if (t > 2)

count++;

sl = strtok(NULL, " .,-:");

}

printf("\nКількість слів - %d", count);

}

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char a[1000];

gets\_s(a);

zxc(a, 0);

return 0;

}



**Завдання 2**: Дано рядок зі стандартним набором символів-роздільників між словами. Необхідно:

1) перевірити, чи є у рядку слова паліндроми;

2) визначити кількість повторень кожного слова;

3) знайти слово, що повторюється в рядку максимальну кількість разів;

4) видалити із рядка повторення слів, сформувати рядок із різних слів;

5) підрахувати у рядку кількість слів, що починаються з однаковій літери;

6) переставити слова за спаданням кількості голосних букв;

7) переставити слова в алфавітному порядку;

8) знайти пари слів, довжини яких кратні.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <windows.h>

#include<stdio.h>

#include<math.h>

#include <conio.h>

#include <stdlib.h>

#include<string.h>

#include<ctype.h>

void zxc(char a[], int count) {//start 1

printf("\nЗавдання 1");

char\* sl, b[1000], c[1000];

strcpy(c, a);

sl = strtok(c, " .,-:");

while (sl != NULL) {

strcpy(b, sl);

\_strrev(b);

if (strcmp(sl, b) == 0)

count++;

sl = strtok(NULL, " .,-:");

}

if (count > 0)

printf("\nКількість слів паліндром - %d\n", count);

if (count == 0)

printf("\nСлів паліндром не знайдено\n");

}//end 1

void pudge(char a[]) {//start 2

printf("\nЗавдання 2\n");

char b[100][100], \* sl, c[100][100], a1[1000];

strcpy(a1, a);

int i = 0, k = 0, p = 0, k1 = 0;

sl = strtok(a1, " .,-:");//створення двухвимірного

while (sl != NULL) {

strcpy(b[i], sl);

i++;

k = i;

sl = strtok(NULL, " .,-:");

}//створив

int u = 0, max = 0, d2[100];

char d1[100][100];

for (i = 0; i < k; i++) {

int count = 1;

strcpy(c[p], b[i]);

p++;

k1 = 0;

for (int st = 0; st < p; st++) {

if (strcmp(b[i], c[st]) == 0) {

k1++;

}

}

if (k1 <= 1) {

for (int j = i + 1; j < k; j++) {

if (strcmp(b[j], b[i]) == 0)

count++;

}

printf("%s --- %d\n", b[i], count);

strcpy(d1[u], b[i]);//3

d2[u] = count;//3

u++;//3

}

}//end 2

int qw;//start 3

printf("\nЗавдання 3\n");

for (int i = 0; i < u; i++) {

if (d2[i] > max) {

max = d2[i];

qw = i;

}

}

printf("Найбільше повторюється слово '%s' --- %d\n", d1[qw], max);//end 3

printf("\nЗавдання 4\n");

int n = 0;//start 4

for (i = 0; i < k - n; i++) {

for (int j = i + 1; j < k - n; j++) {

if (strcmp(b[j], b[i]) == 0) {

n++;

for (p = j; p < k - n; p++)

strcpy(b[p], b[p + 1]);

i = 0;

}

}

}

k = k - n;

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

printf("%s ", b[i]);//end 4

for (i = 0; i < 100; i++)//start clear

strcpy(c[i], "0");

//end clear

p = 1;

printf("\n\nЗавдання 5\n");

for (i = 0; i < k; i++) {//start 5

k1 = 0;

int count = 0;

for (int st = 0; st < p; st++) {

if (strncmp(b[i], c[st], 1) == 0) {

k1++;

}

}

if (k1 == 0) {

strncpy(c[p], b[i], 1);

for (int j = i; j < k; j++) {

if (strncmp(b[i], b[j], 1) == 0)

count++;

}

printf("%s --- %d\n", c[p], count);

p++;

}

}//end 5

for (i = 0; i < 100; i++)//start clear

d2[i] = -1;

//end clear

printf("\nЗавдання 6\n");//start 6

for (i = 0; i < k; i++) {

int count = 0;

for (int j = 0; j < strlen(b[i]); j++) {

if (b[i][j] == 'а' || b[i][j] == 'о' || b[i][j] == 'у' || b[i][j] == 'і' || b[i][j] == 'и' || b[i][j] == 'е' || b[i][j] == 'я'

|| b[i][j] == 'ю' || b[i][j] == 'є' || b[i][j] == 'ї' || b[i][j] == 'a' || b[i][j] == 'e' || b[i][j] == 'i' || b[i][j] == 'o' || b[i][j] == 'u' || b[i][j] == 'y')

{

count++;

}

}

d2[i] = count;

}

int fl;

do

{

fl = 0;

for (int i = 0; i < k; i++)

if (d2[i + 1] > d2[i])

{

k1 = d2[i];

d2[i] = d2[i + 1];

d2[i + 1] = k1;

strcpy(d1[0], b[i]);

strcpy(b[i], b[i + 1]);

strcpy(b[i + 1], d1[0]);

fl = 1;

}

} while (fl);

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

printf("%s ", b[i]);//end 6

printf("\n\nЗавдання 7\n");//star 7

for (i = 0; i < 100; i++)//start clear

d2[i] = -1;

//end clear

for (i = 0; i < k; i++) {

d2[i] = (int)b[i][0];

}

do

{

fl = 0;

for (int i = 0; i < k; i++)

if (d2[i - 1] > d2[i])

{

k1 = d2[i];

d2[i] = d2[i - 1];

d2[i - 1] = k1;

strcpy(d1[0], b[i]);

strcpy(b[i], b[i - 1]);

strcpy(b[i - 1], d1[0]);

fl = 1;

}

} while (fl);

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

printf("%s ", b[i]);//end 7

printf("\n\nЗавдання 8");//start 8

for (i = 0; i < 100; i++)//start clear

d2[i] = -1;

//end clear

for (i = 0; i < k; i++) {

d2[i] = strlen(b[i]);

}

int vt = 0;

for (i = 0; i < k; i++) {

int ki = 0;

if (d2[i] == 1)

vt++;

if (d2[i] != -1 && d2[i] != 1) {

printf("\n");

for (int j = i + 1; j < k; j++) {

if ((!(d2[j] % d2[i]) || !(d2[i] % d2[j])) && d2[j] != -1 && d2[j] != 1) {

if (ki == 0)

printf("%s", b[i]);

ki++;

printf(" - %s", b[j]);

d2[j] = -1;

}

}

}

}

if (vt > 0) {

int ki = 0;

printf("\nСлова нижче кратні до всіх чисел, так як мають довжину 1\n");

for (i = 0; i < k; i++) {

if (d2[i] == 1) {

if (ki == 0)

printf("%s", b[i]);

if (ki > 0)

printf(" - %s", b[i]);

}

}

} //end 8

printf("\n\nКінець :)\n");

}

//ya duzhe lyublyu hraty v dotu tse moya ulyublena ya hra

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char a[1000], b[100][100], \* sl;

gets\_s(a);

for (int i = 0; i < strlen(a); i++) {

if (isupper(a[i]))

a[i] = tolower(a[i]);

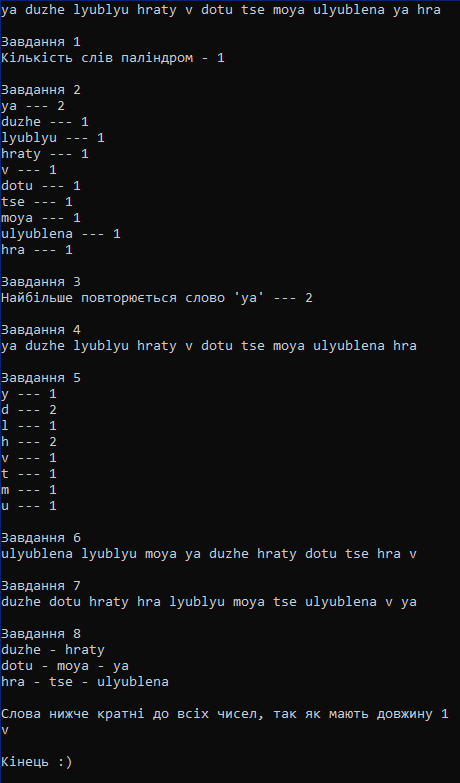
}

zxc(a, 0);

pudge(a);

return 0;

}

****

**Самостійна робота:**

**Завдання 1**: Написати програму, яка з використанням оператора switch перетворюватиме ціле число (від 2 до 5), що вводиться з клавіатури, у відповідний запис («незадовільно», «задовільно», «добре», «відмінно»), в разі введення іншого числа - виводити повідомлення про відсутність такої оцінки.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <windows.h>

#include<stdio.h>

#include<math.h>

#include <conio.h>

#include <stdlib.h>

#include<string.h>

#include<ctype.h>

void spirit() {

for (int i = 0; i != -1;) {

printf("Введіть вашу оцінку, або ведіть '-1' для завершення\n");

scanf\_s("%d", &i);

if ((i < 2 || i > 5) && i != -1) {

printf("Такої оцінки не існує\n");

continue;

}

if(i == -1)

continue;

switch (i) {

case 2:

printf("незадовільно\n");

break;

case 3:

printf("задовільно\n");

break;

case 4:

printf("добре\n");

break;

case 5:

printf("відмінно\n");

break;

default:

break;

}

}

}

int main() {

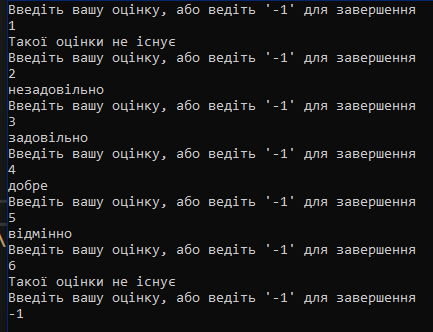
SetConsoleCP(1251);

SetConsoleOutputCP(1251);

spirit();

return 0;

}



**Завдання 2**: Переписати програму із пункту 1 без використання оператора switch, але з використанням масиву рядків.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <windows.h>

#include<stdio.h>

#include<math.h>

#include <conio.h>

#include <stdlib.h>

#include<string.h>

#include<ctype.h>

void spirit() {

char a[5][100];

strcpy(a[0], "незадовільно");

strcpy(a[1], "задовільно");

strcpy(a[2], "добре");

strcpy(a[3], "відмінно");

strcpy(a[4], "Такої оцінки не існує");

for (int i = 0; i != -1;) {

printf("Введіть вашу оцінку, або ведіть '-1' для завершення\n");

scanf\_s("%d", &i);

if (i == -1)

continue;

if ((i < 2 || i > 5) && i != -1) {

puts(a[4]);

continue;

}

puts(a[i - 2]);

}

}

int main() {

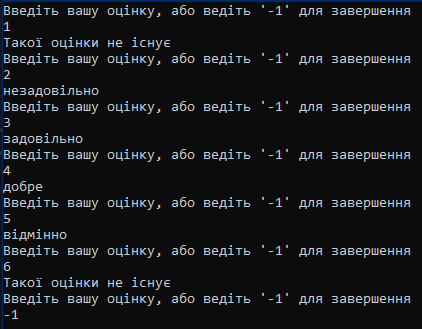
SetConsoleCP(1251);

SetConsoleOutputCP(1251);

spirit();

return 0;

}



**Завдання 3**: Написати програму, яка буде здійснювати зворотне перетворення до завдання №2, тобто перетворювати введену з клавіатури оцінку у вигляді рядка тексту у числове значення.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <windows.h>

#include<stdio.h>

#include<math.h>

#include <conio.h>

#include <stdlib.h>

#include<string.h>

#include<ctype.h>

void spirit() {

char a[6][100], b[100];

strcpy(a[0], "незадовільно");

strcpy(a[1], "задовільно");

strcpy(a[2], "добре");

strcpy(a[3], "відмінно");

strcpy(a[4], "стоп");

strcpy(a[5], "такої оцінки не існує");

for (;;) {

printf("Введіть вашу оцінку, або ведіть 'стоп' для завершення\n");

gets\_s(b);

if (strcmp(a[4], b) == 0)

break;

int k = 0;

for (int i = 0; i < 4; i++) {

if (strcmp(a[i], b) == 0) {

printf("%d\n", i + 2);

k++;

}

}

if (k == 0)

puts(a[5]);

}

}

int main() {

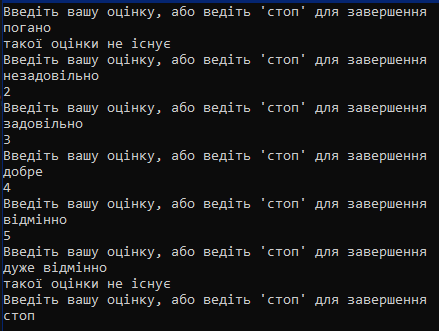
SetConsoleCP(1251);

SetConsoleOutputCP(1251);

spirit();

return 0;

}



**Завдання 4**: Написати програму для обчислення значення арифметичного виразу, яке може містити операції «+», «-», «\*» і «/», операндами яких можуть бути цілі і дійсні числа, а для пріоритету операцій можуть використовуватися круглі дужки.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <windows.h>

#include<stdio.h>

#include<math.h>

#include <conio.h>

#include <stdlib.h>

#include<string.h>

#include<ctype.h>

#define N 100

double spirit(char\* mass)

{

char out[N] = " ", stek[N] = " ";

int v = 0, s = 0;

for (int i = 0; mass[i - 1] != '\0'; i++)

{

if (mass[i] >= '0' && mass[i] <= '9')

{

for (; (mass[i] >= '0' && mass[i] <= '9') || mass[i] == '.'; i++)

{

out[v] = mass[i];

v++;

}

out[v] = ',';

v++;

}

if (mass[i] == '(')

{

stek[s] = mass[i];

s++;

}

if (mass[i] == ')')

{

while (stek[s - 1] != '(')

{

out[v] = stek[s - 1];

v++;

stek[s - 1] = ' ';

s--;

}

stek[s - 1] = ' ';

s--;

}

if (mass[i] == '\*' || mass[i] == '/')

{

while (stek[s - 1] == '\*' || stek[s - 1] == '/')

{

out[v] = stek[s - 1];

v++;

s--;

}

stek[s] = mass[i];

s++;

}

if (mass[i] == '+' || mass[i] == '-')

{

while (stek[s - 1] == '\*' || stek[s - 1] == '/' || stek[s - 1] == '+' || stek[s - 1] == '-')

{

out[v] = stek[s - 1];

v++;

s--;

}

stek[s] = mass[i];

s++;

}

}

for (int i = 0; i < s; i++)

{

out[v] = stek[s - 1 - i];

v++;

}

double result[N];

int ri = 0;

for (int i = 0; i <= v; i++)

{

char chi[10] = "";

int c = 0;

if (out[i] >= '0' && out[i] <= '9')

{

for (; (out[i] >= '0' && out[i] <= '9') || out[i] == '.'; i++)

{

chi[c] = out[i];

c++;

}

double chis = 0;

result[ri] = atof(chi);

ri++;

}

switch (out[i])

{

case '+': {

result[ri - 2] += result[ri - 1];

ri--;

}break;

case '-': {

result[ri - 2] -= result[ri - 1];

ri--;

}break;

case '\*': {

result[ri - 2] \*= result[ri - 1];

ri--;

}break;

case '/': {

result[ri - 2] /= result[ri - 1];

ri--;

}break;

}

}

return result[0];

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char math[N];

gets\_s(math);

double result = spirit(math);

printf("Результат: %.2lf", result);

}



***Висновки:*** я отримав практичні навички роботи з рядками