

МIНIСТЕРСТВО ОСВIТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

**Лабораторна робота №** 2

з дисципліни “ Основи програмування ”

## тема “Обчислення математичних формул. Використання конструкцій розгалуження та циклів ”

|  |  |  |
| --- | --- | --- |
| Виконала  студентка I курсу  групи КП-61  Чумак Надія Русланівна  (*прізвище, ім’я, по батькові*)  варіант №\_\_\_\_\_\_\_ |  | Перевірив  “\_\_\_\_” “\_\_\_\_\_\_\_\_\_\_\_\_” 20\_\_\_ р.  викладач  Гадиняк Руслан Анатолійович  (*прізвище, ім’я, по батькові*) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Штрафні бали:   |  |  | | --- | --- | | **Термін здачі** | **Оформлення звіту** | |  |  | | Нараховані бали:   |  |  |  | | --- | --- | --- | | **Корект. програм (2 бала)** | **Відп. на теор. питання (1 бал)** | **Відп. на прогр. питання (2 бала)** | |  |  |  | | Сумарний бал:   |  | | --- | |  | |

Київ 2016

**Мета роботи**

Навчитися працювати зі статичними масивами різних типів даних мови програмування С.

Застосувати на практиці різні види циклічних конструкцій при роботі з одномірними та багатомірними масивами даних.

Вдосконалити вміння роботи з рядками у мові С.

Навчитися оформлювати консольну програму для зручності роботи користувача.

**Постановка завдання**

#### **Загальні вимоги до завдання**

Всі завдання реалізувати разом у одній консольній програмі.

Після запуску програми користувач має можливість обрати завдання зі списку завдань і перейти у відповідний стан.

Завжди виводити інформацію про стан, у якому знаходиться користувач та надати спеціальну команду (або клавішу) для виклику допомоги (help) по поточному стану.

Розділити консоль у стані на дві "зони": **зону відображення** та **командну зону**.

У зоні відображення виводити об'єкт з яким працюватиме користувач у відповідному стані (наприклад, масив або рядок (string)). При переході до кожного із станів одразу виводити об'єкт у зоні відображення.

У командній зоні дати користувачу можливість вводити команди та виводити результати їх виконання (наприклад, максимальне значення у масиві як результат відповідної команди).

Дати можливість користувачу за допомогою спеціальної команди (або нажаття спеціальної клавіші) перейти у початковий стан, у якому можна обрати інше завдання.

Весь ввід користувача перевіряти на валідність.

Можливі **додаткові бали** за реалізацію підсвічування фону шуканих елементів, або елементів, що змінюються.

**Завдання 1 Одномірний масив**

 У цьому стані у користувача є одномірний масив цілих чисел (розміром 10 елементів) ініціалізований нулями.

Доступні операції над масивом:

1. Заповнити масив випадковими числами від -100 до 100.
2. Обнулити всі елементи масиву.
3. Знайти мінімальний елемент та його індекс.
4. Знайти суму елементів масиву.
5. Вивести добуток від'ємних елементів масиву.
6. Знайти перший елемент, що повторюється найбільшу кількість раз.
7. Поміняти місцями максимальний і мінімальний елементи масиву.
8. Помножити всі елементи масиву на введене число.

**Завдання 2 Два одномірні масиви**

У цьому стані у користувача є 2 одномірних масива дробових чисел (розміром 10) ініціалізованих нулями.

Доступні операції над масивами:

1. Заповнити масиви випадковими дробовими числами від -20.0 до 20.0.
2. Обнулити всі елементи масивів.
3. Знайти значення нового масиву чисел, елементи якого будуть різницею відповідних елементів першого і другого масиву.
4. Вивести номер масиву, у якому максимальна сума елементів.
5. Поміняти місцями мінімальний елемент першого масиву і максимальний елемент другого масиву.

**Завдання 3 Двомірний масив**

У цьому стані у користувача є двомірний масив цілих чисел (розміром 8х8) ініціалізований нулями.

Доступні операції над матрицею:

1. Заповнити масив випадковими числами від -10 до 10.
2. Обнулити всі елементи масиву.
3. Знайти максимальний елемент та його індекси.
4. Знайти суму побічної діагоналі масиву.
5. Знайти суму стовпця за заданим індексом.
6. Поміняти місцями максимальний і мінімальний елементи масиву.
7. Змінити значення елементу за вказаними індексами на задане.

**Завдання 4 Обробка рядків мовою С**

У цьому стані у користувача є буфер (розміром 80 символів) для зберігання строки. У початковому стані буфер повинен містити будь-яку видиму строку.

Доступні операції над рядком:

1. Заповнити строку вводом із консолі.
2. Очистити строку.
3. Вивести довжину строки.
4. Вивести підстроку із заданої позиції і заданої довжини.
5. Вивести список підстрок, розділених заданим символом.
6. Вивести найкоротше слово (слова - непуста послідовність буквенних символів).
7. Знайти та вивести всі дробові числа, що містяться у строці.
8. Знайти та вивести добуток всіх цілих чисел, що містяться у строці.

**Тексти коду програм**

**Завдання 1.**

|  |
| --- |
| task\_1.c |
| #include "lab\_2.h"  void clearLine(char c,int y,int x,int num,enum conAttribute\_e color) {  int i,j;  conSetAttr(color);  for (j = 0; j < num; j ++) {  for (i = 0; i < x; i ++) {  conMove(y+j,1+i);  putchar(c);  }  }  }  void task\_1(void) {  int number = 0;  int Array[10] = { 0 ,0,0,0,0,0,0,0,0,0};  int length = 10;  int imin = 0;  int imax = 0;  int min = 100;  int max = -100;  int temp = 0;  int Sum = 0;  int Product = 1;  int counter = 0;  int i = 0;  int j = 0;  char keyDown;  int x = 0 ;  int y = 0;  int box\_left = 0;  int box\_top = 5;  int box\_width = 87;  int box\_height = 12;  int x\_width\_name = 87;  int x\_height\_name = 5;  int y\_console\_height = 12;  srand(time(0));  /\*prebox\*/  conSetAttr(BG\_BLACK);  for (x = 0;x < x\_width\_name;x ++) {  for (y = 0; y < x\_height\_name; y ++) {  conMove(y, x);  putchar(' ');  }  }  conMove(3,40);  printf("Task 1\n");  conReset();  conSetAttr(BG\_CYAN);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top ; y < box\_top + box\_height ; y++) {  conMove(y, x);  putchar(' ');  }  }  puts("");  conMove(6,2);  printf ("1.Заповнити масив випадковими числами від -100 до 100\n");  conMove(7,2);  printf ("2.Обнулити всі елементи масиву.\n");  conMove(8,2);  printf ("3.Знайти мінімальний елемент та його індекс.\n");  conMove(9,2);  printf ("4.Знайти суму елементів масиву.\n");  conMove(10,2);  printf ("5.Вивести добуток від'ємних елементів масиву.\n");  conMove(11,2);  printf ("6.Знайти перший елемент, що повторюється найбільшу кількість раз.\n");  conMove(12,2);  printf ("7.Поміняти місцями максимальний і мінімальний елементи масиву\n");  conMove(13,2);  printf ("8.Помножити всі елементи масиву на введене число.\n");  conMove(14,10);  printf ("Натисність '9' ,щоб щоб повернутия до головного меню\n");  conMove(15,10);  printf ("Натисність q ,щоб вийти з програми\n");  conReset();  conSetAttr(BG\_BLACK);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top + box\_height ; y < box\_top + box\_height + y\_console\_height; y++) {  conMove(y, x);  putchar(' ');  }  }  while(keyDown != 'q') {  conMove(17, 3);  conSetAttr(BG\_BLACK);  puts ("Enter your number");  keyDown = conGetChar();  switch (keyDown){  case '1': {  for (i = 0; i < length; i++ ) {  Array[i]= rand()%200 - 100;  printf("\t%i",Array[i]);  }  printf("\n");  }  break;  case '2': {  conMove(17,3);  clearLine(' ',17,box\_width-1,2,BG\_BLACK);  conMove(18,2);  for (i = 0;i < length; i++) {  Array[i] = 0;  printf("\t%i",Array[i]);  }  printf("\n");  }  break;  case '3': {  conMove(19, 3);  for(i = 0; i < length; i++) {  if (Array[i] < min) {  min = Array[i];  imin = i;  }  }  printf("\n");  printf("A[%i] = %i\n",imin , min);  }  break;  case '4': {  conMove(20, 3);  for(i = 0;i < length; i++) {  Sum += Array[i];  }  printf("\n");  printf("Sum: %d\n", Sum);  }  break;  case '5': {  conMove(21, 3);  for(i = 0;i < length; i++) {  if (Array[i] < 0) {  Product \*= Array[i];  }  }  printf("\n");  if (Product == 1){  printf("Array doesn`t consist of negative elements\n");  }  else {  printf("Product: %d\n", Product);  }  }  break;  case '6': {  conMove(21, 20);  for( i = 0; i < length; i++) {  counter = 1;  for (j = i + 1; j < length; j++) {  if(Array[i] == Array[j]) {  counter++;  if (counter > max) {  max = counter;  imax = j;  }  }  else {  counter = 0;  }  }  }  if (imax == 0) {  printf("There is no repeating elements\n");  }  else {  printf("First repeating element %d\n",Array[imax]);  }  }  break;  case '7': {  conMove(23, 3);  imax = 0;  imin = 0;  max = -100;  min = 100;  for (i = 0 ; i < length; i ++) {  if (Array[i] < min) {  min = Array[i];  imin = i;  }  if (Array[i] > max) {  max = Array[i];  imax = i;  }  }  temp = Array[imin];  Array[imin]=Array[imax];  Array[imax]=temp;  for (i = 0; i < length; i ++) {  printf("\t%i",Array[i]);  }  printf("\n");  }  break;  case '8': {  conMove(24, 3);  printf ("Enter your number\n");  number = getInt();  conMove(24, 3);  clearLine(' ',24,box\_width-1,2,BG\_BLACK);  conMove(24, 3);  for (i = 0;i < length; i++) {  Array[i] = number \* Array[i];  printf("\t%i",Array[i]);  }  conMove(17, 3);  }  break;  case '9':{  conReset();  menu();  break;  }  }  conReset();  }  } |

**Завдання 2.**

|  |
| --- |
| task\_2.c |
| #include "lab\_2.h"  void clearLine1(char c,int y,int x,int num,enum conAttribute\_e color) {  int i,j;  conSetAttr(color);  for (j = 0; j < num; j ++) {  for (i = 0; i < x; i ++) {  conMove(y+j,1+i);  putchar(c);  }  }  }  void task\_2(void) {  double arr1[10] = { 0 };  double arr2[10] = { 0 };  double arr3[10];  int length = 10;  double Sum1 = 0;  double Sum2 = 0;  int i = 0;  int imin = 0;  int imax = 0;  double dmin = 20.0;  double dmax = -20.0;  int temp = 0;  char keyDown = 0;  int x ;  int y ;  int box\_left = 0;  int box\_top = 5;  int box\_width = 87;  int box\_height = 12;  int x\_width\_name = 87;  int x\_height\_name = 5;  int y\_console\_height = 12;  /\*prebox\*/  conSetAttr(BG\_BLACK);  for (x = 0;x < x\_width\_name;x ++) {  for (y = 0; y < x\_height\_name; y ++) {  conMove(y, x);  putchar(' ');  }  }  conMove(3,40);  printf("Task 2\n");  conReset();  conSetAttr(BG\_CYAN);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top ; y < box\_top + box\_height ; y++) {  conMove(y, x);  putchar(' ');  }  }  puts("");  conMove(6,2);  printf ("1.Заповнити масиви випадковими дробовими числами від -20.0 до 20.0\n");  conMove(7,2);  printf ("2.Обнулити всі елементи масивів.\n");  conMove(8,2);  printf ("3.Знайти значення нового масиву чисел, елементи якого будуть різницею \n відповідних елементів першого і другого масиву.\n");  conMove(10,2);  printf ("4.Вивести номер масиву, у якому максимальна сума елементів.\n");  conMove(11,2);  printf ("5.Поміняти місцями мінімальний елемент першого масиву і максимальний елемент\n другого масиву.\n");  conMove(14,10);  printf ("Натисність '9' ,щоб щоб повернутия до головного меню\n");  conMove(15,10);  printf ("Натисність q ,щоб вийти з програми\n");  conReset();  conSetAttr(BG\_BLACK);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top + box\_height ; y < box\_top + box\_height + y\_console\_height; y++) {  conMove(y, x);  putchar(' ');  }  }  while(keyDown != 'q') {  conMove(17, 3);  conSetAttr(BG\_BLACK);  puts ("Enter your number");  keyDown = conGetChar();  switch (keyDown) {  case '1': {  for (i = 0;i < length; i++) {  arr1[i] = (rand()%80)/2.0-40/2.0;  printf("\t%0.1f",arr1[i]);  }  printf("\n");  for (i = 0;i < length; i++) {  arr2[i] = (rand()%80)/2.0-40/2.0;  printf("\t%0.1f",arr2[i]);  }  printf("\n");  break;  }  case '2': {  clearLine1(' ',17,box\_width-1,3,BG\_BLACK);  conMove(18, 3);  for (i = 0;i < length; i++) {  arr1[i] = 0;  printf("\t%0.1f",arr1[i]);  }  printf("\n");  for (i = 0;i < length; i++) {  arr2[i] = 0;  printf("\t%0.1f",arr2[i]);  }  printf("\n");  }  break;  case '3': {  conMove(20, 3);  for (i = 0;i < length; i++) {  arr3[i] = arr1[i]- arr2[i];  printf("\t%0.1f",arr3[i]);  }  printf("\n");  }  break;  case '4': {  conMove(21, 3);  for(i = 0;i < length; i++) {  Sum1 += arr1[i];  }  for(i = 0;i < length; i++) {  Sum2 += arr2[i];  }  if (Sum1 > Sum2) {  printf("1 - number of array1\n");  } else if (Sum2 > Sum1) {  printf("2 - number of array2\n");  }  else {  printf("arrays are equal\n");  }  }  break;  case '5': {  conMove(22, 3);  for (i = 0;i < length;i ++) {  if (arr1[i] < dmin) {  dmin = arr1[i];  imin = i;  }  if (arr2[i] > dmax) {  dmax = arr2[i];  imax = i;  }  }  temp = arr1[imin];  arr1[imin] = arr2[imax];  arr2[imax] = temp;  for (i = 0; i < length; i ++) {  printf("\t%0.1f",arr1[i]);  }  printf("\n");  for (i = 0; i < length; i ++) {  printf("\t%0.1f",arr2[i]);  }  printf("\n");  }  break;  case '9': {  conReset();  menu();  }  break;  }  }  conReset();  } |

**Завдання 3.**

|  |
| --- |
| task\_3.c |
| #include "lab\_2.h"  void clearLine2(char c,int y,int x,int num,enum conAttribute\_e color) {  int i,j;  conSetAttr(color);  for (j = 0; j < num; j ++) {  for (i = 0; i < x; i ++) {  conMove(y+j,1+i);  putchar(c);  }  printf("\n");  }  }  void task\_3(void)  {  char keyDown ;  int i = 0;  int j = 0;  int max = -10;  int min = 10;  int imin = 0;  int jmin = 0;  int imax = 0;  int jmax = 0;  int temp = 0;  int length = 8;  int Sum = 0;  int x ;  int y ;  int box\_left = 0;  int box\_top = 5;  int box\_width = 87;  int box\_height = 12;  int x\_width\_name = 87;  int x\_height\_name = 5;  int y\_console\_height = 12;  int value\_element = 0;  int i\_element = 0;  int j\_element = 0;  int arr4[8][8]= {{0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}} ;  srand(time(0));  /\*prebox\*/  conSetAttr(BG\_BLACK);  for (x = 0;x < x\_width\_name;x ++) {  for (y = 0; y < x\_height\_name; y ++) {  conMove(y, x);  putchar(' ');  }  }  conMove(3,40);  printf("Task 3\n");  conReset();  conSetAttr(BG\_CYAN);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top ; y < box\_top + box\_height ; y++) {  conMove(y, x);  putchar(' ');  }  }  puts("");  conMove(6,2);  printf ("1.Заповнити масив випадковими числами від -10 до 10\n");  conMove(7,2);  printf ("2.Обнулити всі елементи масиву.\n");  conMove(8,2);  printf ("3.Знайти максимальний елемент та його індекси.\n");  conMove(9,2);  printf ("4.Знайти суму побічної діагоналі масиву.\n");  conMove(10,2);  printf ("5.Знайти суму стовпця за заданим індексом.\n");  conMove(11,2);  printf ("6.Поміняти місцями максимальний і мінімальний елементи масиву.\n");  conMove(12,2);  printf ("7.Змінити значення елементу за вказаними індексами на задане.\n");  conMove(14,10);  printf ("Натисність '9' ,щоб щоб повернутия до головного меню\n");  conMove(15,10);  printf ("Натисність q ,щоб вийти з програми\n");  conSetAttr(BG\_BLACK);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top + box\_height ; y < box\_top + box\_height + y\_console\_height; y++) {  conMove(y, x);  putchar(' ');  }  }  while (keyDown != 'q') {  conMove(17, 1);  conSetAttr(BG\_BLACK);  puts ("Enter your number");  keyDown = conGetChar();  switch (keyDown){  case '1': {  for (i = 0; i < length; i++) {  for(j = 0; j < length; j ++) {  arr4[i][j] =rand()%20 - 10;  printf("\t%3i",arr4[i][j]);  }  printf("\n");  }  printf("\n");  }  break;  case '2': {  clearLine2(' ',17,box\_width-1,7,BG\_BLACK);  conMove(18, 3);  for (i = 0; i < length; i++) {  for(j = 0; j < length; j ++) {  arr4[i][j] = 0;  printf("\t%3i",arr4[i][j]);  }  printf("\n");  }  printf("\n");  }  break;  case '3': {  conMove(26,3);  for (i = 0; i < length; i++) {  for(j = 0; j < length; j ++) {  if ( arr4[i][j] > max) {  max = arr4[i][j];  imax = i;  jmax = j;  }  }  }  printf("Array[%i][%i] = %i\n",imax , jmax,max);  }  break;  case '4': {  clearLine2(' ',26,box\_width-1,1,BG\_BLACK);  conMove(26,3);  for (i = 0; i < length; i++) {  for(j = 0; j < length; j ++) {  if ( i + j == length -1) {  Sum += arr4[i][j];  }  }  }  printf("Sum: %d\n", Sum);  }  break;  case '5': {  Sum = 0;  clearLine2(' ',26,box\_width-1,1,BG\_BLACK);  conMove(26,3);  puts ("Введіть цілочисельне j:");  j = getInt();  conMove(28,3);  if (j >= 0 && j < 8) {  for(i = 0; i < length; i ++) {  Sum += arr4[i][j];  }  printf("Sum: %d\n", Sum);  }  else {  printf("Index is not found\n");  }  clearLine2(' ',29,box\_width-1,1,BG\_DEFAULT);  conMove(17,3);  }  break;  case '6': {  clearLine2(' ',17,box\_width-1,14,BG\_BLACK);  conMove(18,1);  max = -10;  imax = 0;  jmax = 0;  for (i = 0; i < length; i++) {  for(j = 0; j < length; j ++) {  if ( arr4[i][j] > max) {  max = arr4[i][j];  imax = i;  jmax = j;  }  if (arr4[i][j] < min) {  min = arr4[i][j];  imin = i;  jmin = j;  }  }  }  temp = arr4[imin][jmin];  arr4[imin][jmin] = arr4[imax][jmax];  arr4[imax][jmax] = temp;  for (i = 0; i < length; i++) {  for(j = 0; j < length; j ++) {  printf("\t%3i",arr4[i][j]);  }  printf("\n");  }  }  break;  case '7': {  clearLine2(' ',16,box\_width-1,14,BG\_BLACK);  conMove(19,1);  puts ("Enter i:");  i\_element = getInt();  puts ("Enter j:");  j\_element = getInt();  conMove(23,1);  puts ("Enter value of your element:");  value\_element = getInt();  clearLine2(' ',17,box\_width-1,11,BG\_BLACK);  conMove(18,3);  if (i\_element >= 0 && j\_element >= 0 && i\_element < 8 && j\_element < 8) {  for (i = 0; i < length; i++) {  for(j = 0; j < length; j ++) {  if (i == i\_element && j == j\_element) {  arr4[i][j] = value\_element;  printf("\t%3i",arr4[i][j]);  } else {  printf("\t%3i",arr4[i][j]);  }  }  printf("\n");  }  }  else {  printf("Try to enter index from 0 to 7");  }  }  break;  case '9': {  conReset();  menu();  }  break;  }  }  conReset();  } |

**Завдання 4.**

|  |
| --- |
| task\_4.c |
| #include "lab\_2.h"  void clearLine3(char c,int y,int x,int num,enum conAttribute\_e color) {  int i,j;  conSetAttr(color);  for (j = 0; j < num; j ++) {  for (i = 0; i < x; i ++) {  conMove(y+j,1+i);  putchar(c);  }  }  }  void getDoubles(char \* p){  char \* t;  char isRealPresent = 0;  for(; \*p != '\0'; p++) {  if(isdigit(\*p) || (\*p=='-' && isdigit(\*(p+1)))) {  for(t = p + 1; \*t != '\0' && \*t != '.' && isdigit(\*t); t++) ;  if(\*t == '.' && isdigit(\*(t + 1))) {  isRealPresent = 1;  printf("%.3f ", atof(p));  for(p = t+1; isdigit(\*(p + 1)); p++) ;  }  }  }  if(!isRealPresent) {  printf("There is no float numbers" );  }  }  void task\_4(void) {  char str[81] = "\*\*\*\*\*\*Hi,I am a string !\*\*\*\*\*\*";  char character = '\0';  int length = 0;  char keyDown ;  int i = 0;  int x1 = 0;  int x2 = 0;  int count = 0;  int imin = 0;  int min = INT\_MAX;  long product = 1;  int countdigit = 0;  int x ;  int y ;  int box\_left = 0;  int box\_top = 5;  int box\_width = 87;  int box\_height = 12;  int x\_width\_name = 87;  int x\_height\_name = 5;  int y\_console\_height = 12;  /\*prebox\*/  conSetAttr(BG\_BLACK);  for (x = 0;x < x\_width\_name;x ++) {  for (y = 0; y < x\_height\_name; y ++) {  conMove(y, x);  putchar(' ');  }  }  conMove(3,40);  printf("Task 4\n");  conReset();  conSetAttr(BG\_CYAN);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top ; y < box\_top + box\_height ; y++) {  conMove(y, x);  putchar(' ');  }  }  puts("");  conMove(6,2);  printf ("1.Заповнити строку вводом із консолі.\n");  conMove(7,2);  printf ("2.Очистити строку.\n");  conMove(8,2);  printf ("3.Вивести довжину строки.\n");  conMove(9,2);  printf ("4.Вивести підстроку із заданої позиції і заданої довжини.\n");  conMove(10,2);  printf ("5.Вивести список підстрок, розділених заданим символом.\n");  conMove(11,2);  printf ("6.Вивести найкоротше слово .\n");  conMove(12,2);  printf ("7.Знайти та вивести всі дробові числа, що містяться у строці.\n");  conMove(13,2);  printf ("8.Знайти та вивести добуток всіх цілих чисел, що містяться у строці.\n");  conMove(14,10);  printf ("Натисність '9' ,щоб щоб повернутия до головного меню\n");  conMove(15,10);  printf ("Натисність q ,щоб вийти з програми\n");  conSetAttr(BG\_BLACK);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top + box\_height ; y < box\_top + box\_height + y\_console\_height; y++) {  conMove(y, x);  putchar(' ');  }  }  while (keyDown != 'q') {  conMove(17, 3);  conSetAttr(BG\_BLACK);  puts ("Enter your number");  keyDown = conGetChar();  switch(keyDown) {  case '1': {  conMove(19, 1);  puts("Please enter a string (maximum 80 characters):");  fgets(str,81, stdin);  puts("");  }  break;  case '2': {  str[0] = '\0';  length = strlen(str);  for (i = 0; i < length;i ++ ) {  printf("%i",str[0]);  }  }  break;  case '3': {  conMove(22,1);  length = strlen(str);  if (length == 80 || length == 0){  printf("Length = %d\n", length);  }  else {  printf("Length = %d\n", length - 1);  }  }  break;  case '4': {  conMove(23,1);  length = strlen(str);  if (length == 80 || length == 0){  length = strlen(str);  }  else {  length = strlen(str) - 1;  }  puts("Enter start position of your stringle:");  x1 = getInt();  puts("Enter length of your string:");  x2 = getInt();  if (x2 < length - x1) {  for (i = x1; i < (x1+ x2); i ++) {  putchar(str[i]);  }  puts("");  } else {  printf("There is no string or you puts an alphabetic numbers.Try again:)\n");  }  }  break;  case '5': {  clearLine3(' ',21,box\_width-1,7,BG\_BLACK);  conMove(22,3);  printf("Enter some characters\n");  character = conGetChar();  for(i = 0; str[i] != '\0'; i++) {  if(str[i] == character) {  printf("\n");  } else {  putchar(str[i]);  }  }  }  break;  case '6': {  clearLine3(' ',22,box\_width-1,8,BG\_BLACK);  conMove(21,1);  for (i = 0; str[i] != '\0'; i ++) {  if (isalpha(str[i])) {  count ++;  } else {  if (count < min) {  min = count;  imin = i - count;  }  count = 0;  }  }  for ( i = imin;i < imin + min; i++){  putchar(str[i]);  }  if (min == 0) {  printf("The string does not consist of words");  }  }  break;  case '7': {  clearLine3(' ',22,box\_width-1,7,BG\_BLACK);  conMove(23,1);  /\*for (i = 0; str[i] != '\0'; i++) {  char ch = str[i];  char nextCh = str[i + 1];  if (isdigit(ch) || ('-' == ch && isdigit(nextCh))){  double number = atof(str + i);  count = printf("%f", number);  puts("");  i += count - 1;  countdigit++;  }  }  if (countdigit == 0) {  printf("Input string don't contain digits\n");  }\*/  getDoubles(&str[0]);  putchar('\n');  }  break;  case '8': {  clearLine3(' ',22,box\_width-1,7,BG\_BLACK);  conMove(22,1);  for (i = 0; str[i] != '\0'; i++) {  char ch = str[i];  char nextCh = str[i + 1];  if (isdigit(ch) || ('-' == ch && isdigit(nextCh)) ){  int number = atoi(str + i);  int count = printf("%d", number);  puts("");  product \*= number;  i += count - 1;  countdigit ++;  }  }  if (countdigit == 0) {  printf("Input string don't contain digits\n");  } else {  printf("Mult of digits in string: %ld\n",product);  }  }  break;  case '9': {  conReset();  menu();  }  break;  }  }  conReset();  } |

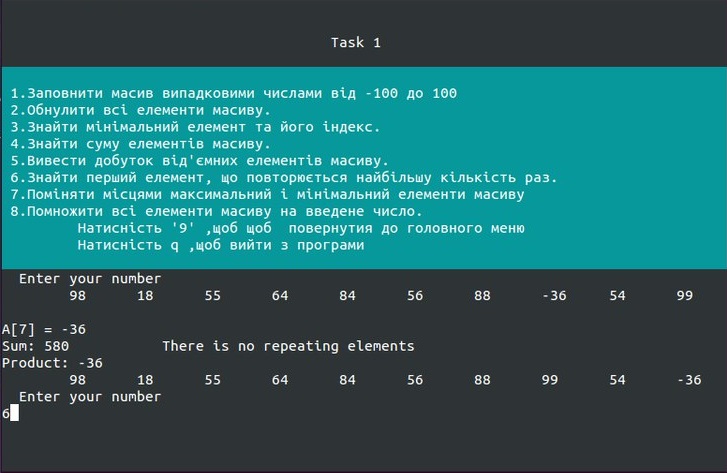
|  |
| --- |
| lab\_2.h |
| #include <stdio.h>  #include <stdlib.h>  #include <time.h>  #include <progbase.h>  #include <pbconsole.h>  #include <math.h>  #include <string.h>  #include <limits.h>  #include <ctype.h>  void task\_1(void);  void task\_2(void);  void task\_3(void);  void task\_4(void);  void menu(void); |

|  |
| --- |
| menu.c |
| #include "lab\_2.h"  void menu(void) {  char keyDown ;  int x = 0 ;  int y = 0;  int box\_left = 0;  int box\_top = 5;  int box\_width = 87;  int box\_height = 12;  int x\_width\_name = 87;  int x\_height\_name = 5;  int y\_console\_height = 12;  conClear();  /\*prebox\*/  conSetAttr(BG\_BLACK);  for (x = 0;x < x\_width\_name;x ++) {  for (y = 0; y < x\_height\_name; y ++) {  conMove(y, x);  putchar(' ');  }  }  conReset();  conSetAttr(BG\_BLACK);  conMove(3,35);  printf(" MAIN MENU\n");  conReset();  conSetAttr(BG\_CYAN);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top ; y < box\_top + box\_height ; y++) {  conMove(y, x);  putchar(' ');  }  }  puts("");  conMove(6,2);  printf ("1.Одномірний масив\n");  conMove(7,2);  printf ("2.Два одномірні масиви\n");  conMove(8,2);  printf ("3.Двомірний масив\n");  conMove(9,2);  printf ("4.Обробка рядків мови С\n");  conMove(15,10);  printf ("Натисність q ,щоб вийти з програми\n");  conReset();  conSetAttr(BG\_BLACK);  for ( x = box\_left;x < box\_left + box\_width; x++) {  for (y = box\_top + box\_height ; y < box\_top + box\_height + y\_console\_height; y++) {  conMove(y, x);  putchar(' ');  }  }  while (keyDown != 'q') {  conMove(17, 3);  puts ("Enter your number:");  keyDown = conGetChar();  switch (keyDown) {  case '1':  task\_1();  break;  case '2':  task\_2();  break;  case '3':  task\_3();  break;  case '4':  task\_4();  break;  }  }  conReset();  conMove(32,1);  } |

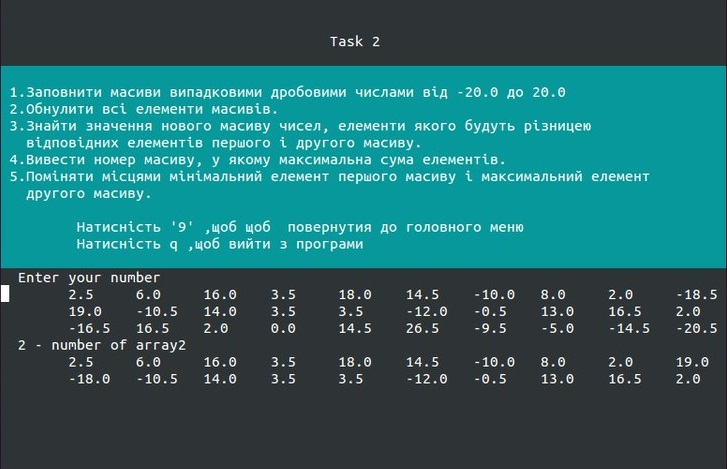
|  |
| --- |
| main.c |
| #include "lab\_2.h"  int main() {  srand((unsigned)time(NULL));  menu();  conReset();  return 0;  } |

**Приклади результатів**

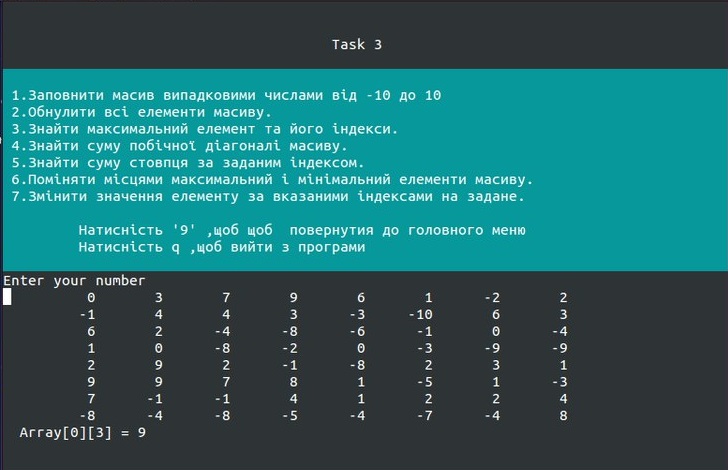
**Завдання 1.**



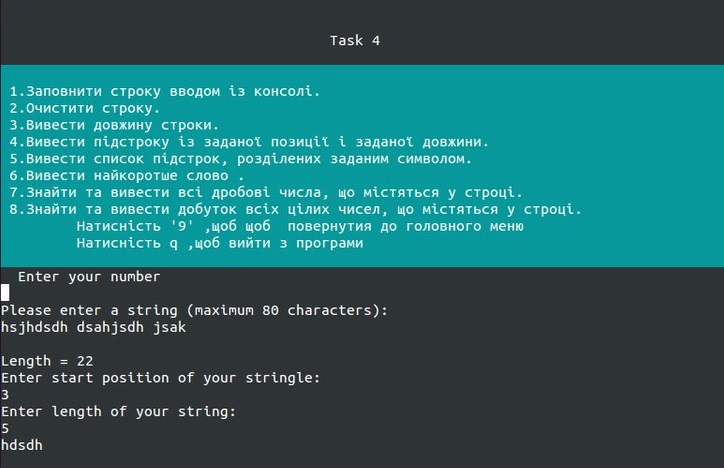
**Завдання 2.**



**Завдання 3.**



**Завдання 4.**



**Висновки**

На цій лабораторній роботі ми навчилися працювати зі статичними масивами різних типів даних мови програмування С,

застосувати на практиці різні види циклічних конструкцій при роботі з одномірними та багатомірними масивами даних,

вдосконалити вміння роботи з рядками у мові С,

навчилися оформлювати консольну програму для зручності роботи користувача.