НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

імені Ігоря Сікорського»

Факультет прикладної математики

Кафедра прикладної математики

Звіт

із лабораторної роботи №*1*

із дисципліни «Основи програмування»

на тему

«Базові типи даних, введення-виведення,  
бітові операції, операції зсуву»

|  |  |
| --- | --- |
| Виконав: | Керівник: |
| студент групи КМ-73 | *асистент Громова В.В.* |
| *Садченко М.В.* |  |

Київ — 2018

ЗМІСТ

[ВСТУП 3](#_Toc506867314)

[1 ПОСТАНОВКА ЗАДАЧІ 3](#_Toc506867315)

[1.1 Мета роботи 3](#_Toc506867316)

[1.2 Варіант та завдання лабораторної роботи 3](#_Toc506867317)

[2 ТЕОРЕТИЧНІ ВІДОМОСТІ 4](#_Toc506867318)

[2.1 Типи даних у мові C 4](#_Toc506867319)

[2.2 Константи у мові C 4](#_Toc506867320)

[2.3 Деякі функції стандартного введення-виведення 5](#_Toc506867321)

[2.4 Бітові операції та операції зсуву 5](#_Toc506867322)

[3. ОПИС ПРОГРАМИ 6](#_Toc506867323)

[3.1 Опис першої програми 6](#_Toc506867324)

[3.2 Опис другої програми 7](#_Toc506867325)

[3.3 Опис бібліотек та функцій 7](#_Toc506867326)

[ВИСНОВКИ 8](#_Toc506867327)

[ПЕРЕЛІК ПОСИЛАНЬ 8](#_Toc506867328)

[ДОДАТОК А (Відповіді на контрольні запитання) 9](#_Toc506867329)

[ДОДАТОК Б (Код програми) 11](#_Toc506867330)

[Перша програма 11](#_Toc506867331)

[Друга програма 13](#_Toc506867332)

[ДОДАТОК В (Результат роботи програми) 15](#_Toc506867333)

[Перша програма 16](#_Toc506867334)

[Друга програма 17](#_Toc506867335)

# 

# ВСТУП

Даний звіт створено для детального ознайомлення з лабораторною роботою №1, її завданням, структурою, розробкою програмного коду мовою програмування C, у супроводі детального розбору поставлених вимог.

# 1 ПОСТАНОВКА ЗАДАЧІ

## 1.1 Мета роботи

Здобути практичні навички роботи з типами даних мови програмування С, бітовими операціями, використання функцій стандартного уведення-виведення.

## 1.2 Варіант та завдання лабораторної роботи

Було виконано роботу за варіантом №24. Були виконані наступні

завдання:

**Завдання №1**:

1. Обчислити поверхню усіченого конуса s=π(R2 + (R+r)L+ r2) і об'єм v=1/3π(R2+r2+Rr)h.
2. Обчислити периметр і площу трикутника, заданого координатами своїх вершин.

**Завдання №2**:

Заголовок пакета в системі передачі даних має формат:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № розрядів | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| значення | T | T | T | 0 | S | S | S | S | L | L | L | L | L | L | L | L |

де TTT— тип пакета;

S..S — ідентифікатор джерела;

L..L — довжина пакета.

# 2 ТЕОРЕТИЧНІ ВІДОМОСТІ

## 2.1 Типи даних у мові C

|  |  |  |
| --- | --- | --- |
| Тип | Розмір, біт | Діапазон значень |
| char | 8 | -128...127 |
| unsigned char | 8 | 0...255 |
| enum | 16 | -32768...32767 |
| short | 16 | -32768...32767 |
| unsigned short | 16 | 0...65535 |
| int | 16 | -32768...32767 |
| unsigned int | 16 | 0...65535 |
| long | 32 | -2147483648...2147483647 |
| unsigned long | 32 | 0...4294967295 |
| float | 32 | 3,4e-38...3,4e+38 |
| double | 64 | 1,7е-308...1,7е+308 |
| longdouble | 80 | 3,4е-4932...1,1е+4932 |

## 2.2 Константи у мові C

*Константа* являє собою мовну конструкцію, що позначає зображення фіксованого числового, рядкового або символьного значення. Їх розділяють на п’ять груп: цілі, дійсні, перелічувані, символьні, рядкові. Перелічувані константи зазвичай відносять до цілочисельного типу даних.

*Рядок* являє собою послідовність символів, оточену лапками. Розміщуючи рядок в пам’яті, компілятор автоматично додає в його кінець нульовий символ ('\0').

## 2.3 Деякі функції стандартного введення-виведення

Функція printf() застосовується для виведення інформації, scanf()- для введення.

Нижче перераховано основні специфікатори:

* %с — одинарний символ;
* %d — десяткове ціле число зі знаком;
* %f — число з плаваючою крапкою (десяткове подання);
* %s — рядок символів (для рядкових змінних);
* %u — десяткове ціле число без знаку;
* %% — друк знаку відсотка.

Функція putchar() слугує для виведення одного символу на екран.

У випадку успішного виконання функція повертає цей же код, неуспішного — константу EOF.

Функція getchar() слугує для уведення одного символу з клавіатури.

Функція puts()слугує для виведення рядка символів на екран.

Функція gets() слугує для уведення рядка символів із клавіатури.

## 2.4 Бітові операції та операції зсуву

У мові С наявні такі бітові операції:

* ~ — бітове заперечення (одномісна);
* & — побітове І (двомісна);
* ^ — побітове АБО (двомісна);
* | — побітове ЧИ (двомісна).

До бітових операцій належать *операції зсуву* << і >>, які дають змогу зсунути всі біти числа вліво і вправо на задану кількість розрядів, відповідно:

* a << b — зсув бітів змінної a вліво на b позицій;
* a >> b — зсув бітів змінної a вправо на b позицій.

Усі бітові операції виконують зліва направо. У наступному рядку наведено бітові операції в порядку зменшення їхнього пріоритету:

~, <<, >>, &, ^, |.

Для двомісних бітових операцій визначено операції присвоювання:

* a <<= b еквівалентно a = a << b,
* a >>= b еквівалентно a = a >> b,
* a &= b еквівалентно a = a & b,
* a ^= b еквівалентно a = a ^ b,
* a |= b еквівалентно a = a | b.

# 3. ОПИС ПРОГРАМИ

## 3.1 Опис першої програми

Спочатку на екрані з’являється так звана «шапка» та варіанти натискання горячих клавіш. Користувачеві надається можливість обрати між запуском першої, другої підпрограм, очисткою консолі та виходом. Реалізовано за допомогою циклу do-while та switch-case.

Натиснувши «1», запуститься перша підпрограма, де потрібно ввести дані про змінні з формул обчислення периметру та площі усіченого конуса. Кожний ввід контролюється за допомогою валідації, що знаходиться в окремій функції та яка дозволяє пройти лише дійсному числу. Після обчислень на екран виведеться результат.

За запуском другої підпрограми (кнопка «2») теж слідує ввід координат типу float, що контролюється валідацією. Для допомоги обчислень було застосовано методи sqrt() та pow() з бібліотеки <math.h>. Після обрахувань сторін трикутника за координатами, а далі й периметра і площі, користувачеві виведеться результат.

## 3.2 Опис другої програми

У другій програмі теж виводиться юзеру варіанти доступу до двух підпрограм, очистки консолі та виходу з програми, відмінностей у реалізації немає.

При виборі клавіші «1», потрапляємо на початок першої підпрограми, де необхідно ввести бітові дані шістнадцятковими числами. Вхідними даними є: тип та довжина пакета, ідентифікатор джерела. Для відбору шістнадцядкових чисел застосовується окрема валідація. Обрахування «слова» виконується за допомогою операцій зсуву та бітових операцій. Після цього на екран виводиться чотирьохзнакове шістнадцяткове число.

Друга підпрограма є протилежною до першої. Тобто в першій ми шукали «слово» з введених даних, а тут навпаки – необхідно знайти початкові дані зі «слова». Після вводу користувачем 4-знакового числа на екран виведеться значення типу та довжини пакета, ідентифікатора джерела.

## 3.3 Опис бібліотек та функцій

Для початку, опишемо використані бібліотеки у програмі:

<stdio.h> - відповідає за ввод\вивід даних;

<stdlib.h> - відповідає за функції, що займаються виділенням пам'яті, контроль процесу виконання програми, перетворення типів та інші;

<math.h> - надає прототипи функцій, розроблених для виконання простих математичних операцій;

<string.h> - містить функції для роботи з нуль-термінованими рядками і різними функціями роботи з пам'яттю.

<ctype.h> - містить оголошення функцій для класифікації символів.

<conio.h> - застосовується для створення текстового інтерфейсу.

Опис функцій:

Програма починається з виконання функціїї int main(), яка визиває валідацію (у першій підпрограмі 1 та 2, у другій – 3):

1. float valid\_float()
2. int valid\_int()
3. int valid\_hex()

# ВИСНОВКИ

Під час виконання лабораторної роботи були вивчені базові типи даних у мові програмування С. Також ми ознайомились з операціями вводу/виводу даних та з логічними бітовими операціями та операціями зсуву.

# ПЕРЕЛІК ПОСИЛАНЬ

1. Програмування мовою С. Методичні вказівки до виконання лабораторних робіт із кредитного модуля «Програмування-2. С.» для студентів спеціальності 113 «Прикладна математика» [Текст] / В. В. Громова. — К. : НТУУ «КПІ», 2016. —5- 22 с.

# ДОДАТОК А (Відповіді на контрольні запитання)

1. *Перерахуйте базові типи даних у мові С.*

Float, char, unsigned, enum, short, int, long, double float, longdouble float.

1. *Визначте тип перерахованих констант: 102, 0675, 0Xff4, 067L.*

102 – цілочисельна десяткова константа, 0675 – цілочисельна вісімкова константа, 0xff4 – цілочисельна шістнадцяткова константа, 067L – дійсна константа longdouble.

1. *Запишіть директиву #define для задання константи з іменем LENGTH, яка дорівнює 10.*

#define LENGTH 10

1. *Стандартні функції уведення та виведення у мові С.*

Функція printf() повертає або число відображених символів, або від’ємне число (у випадку некоректної роботи). Функція gets() слугує для уведення рядка символів із клавіатури. Функція getchar() слугує для уведення одного символу з клавіатури. Для уведення інформації з клавіатури зручно використовувати функцію scanf().

1. *Що являють собою аргументи функцій scanf() і printf()?*

printf(const char\* format*[*,argument*]…*);

int scanf(const char\* format*[*,argument*]…*);

де \*format визначає форматний рядок для визначення типу даних, які уводять, і може містити специфікатор.

1. *Наведіть приклад використання функції printf() для виведення значень двох цілочисельних змінних на екран.*

printf(“Number is : %d”, 23);

1. *Запишіть функцію scanf() для уведення символу з клавіатури.*

int number;

scanf(“%d”, &number);

1. *Які особливості виконання функції getchar()?*

Функція getchar() слугує для уведення одного символу з клавіатури. Вона має такий синтаксис:

int getchar(void).

1. *Логічні операції в мові С.*

* ~ — бітове заперечення (одномісна);
* & — побітове І (двомісна);
* ^ — побітове АБО (двомісна);
* | — побітове ЧИ (двомісна).

1. *Операції зсуву в мові С.*

* a << b — зсув бітів змінної a вліво на b позицій;
* a >> b — зсув бітів змінної a вправо на b позицій.

# ДОДАТОК Б (Код програми)

## Перша програма

#include <stdio.h>

#include <string.h> // strlen(), strchr()

#include <ctype.h> // isdigit()

#include <stdlib.h> // atof(), exit(0)

#include <conio.h> // system("cls"), getch()

#include <math.h>

float valid\_float() {

char str[65];

float num;

while (1) {

int smb = 0, dot = 0;

scanf("%s", str); *//input of value*

int len = strlen(str);

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

if (isdigit(str[i]) == 0) {

if (str[i] == '.') {

++dot;

}

else {

++smb;

**break**;

}

}

}

if ((dot < 2) && ((smb == 0) ||

((smb == 1) && (str[0] == '-')))) {

**break**;

}

else printf("Need float number!: ");

}

num = atof(str); *//convert str to float*

return num;

}

int main() {

printf("**\n**Sadchenko Nikita, student of KM-73.**\n**### Laboratory Work 1 ###**\n**");

int choice;

do {

printf("**\n**Press '1' to calculate perimeter and area of truncated cone.**\n**"\

*// Периметр і площа усіченого конуса.*

"Press '2' to calculate perimeter and area of triangle with coordinates.**\n**"\

"Press '3' to clear console.**\n**"\

"Press '9' to get out program.**\n**");

choice = getch();

} while (!strchr("1239", choice));

switch (choice) {

case '1': {

float R, r, L, h, s, v;

printf("**\n**Enter R: ");

do {

R = valid\_float();

} while (R <= 0);

printf("Enter r: ");

do {

r = valid\_float();

} while (r <= 0);

printf("Enter L: ");

do {

L = valid\_float();

} while (L <= 0);

printf("Enter h: ");

do {

h = valid\_float();

} while (h <= 0);

s = M\_PI\*(R\*R + (R + r)\*L + r\*r);

v = M\_PI/3 \* (R\*R + r\*r + R\*r)\*h;

printf("s = %g**\n**", s);

printf("v = %g**\n\n**", v);

**break**;

}

case '2': {

float x1, y1, x2, y2, x3, y3;

float ab, bc, ac, p, s;

printf("**\n**Enter x-coordinate of A: ");

x1 = valid\_float();

printf("Enter y-coordinate of A: ");

y1 = valid\_float();

printf("Enter x-coordinate of B: ");

x2 = valid\_float();

printf("Enter y-coordinate of B: ");

y2 = valid\_float();

printf("Enter x-coordinate of C: ");

x3 = valid\_float();

printf("Enter y-coordinate of C: ");

y3 = valid\_float();

ab = sqrt(pow((x1-x2),2) + pow((y1-y2),2));

bc = sqrt(pow((x2-x3),2) + pow((y2-y3),2));

ac = sqrt(pow((x1-x3),2) + pow((y1-y3),2));

p = ab + bc + ac;

s = sqrt((p/2)\*((p/2) - ab)\*((p/2) - bc)\*((p/2) - ac));

printf("p = %g**\n**", p);

printf("s = %g**\n\n**", s);

**break**;

}

case '3': {

printf(" 7 ballov :)**\n**");

system("cls");

**break**;

}

case '9': exit(0); **break**;

default: **break**;

}

return main();

}

## Друга програма

#include <stdio.h>

#include <string.h> // strlen(), strchr()

#include <ctype.h> // isxdigit()

#include <stdlib.h> // atoi(), strtol(), exit(0)

#include <conio.h> // system("cls"), getch()

int valid\_int() {

char str[65];

int num;

while (1) {

int j = 0;

scanf("%s", str); *//input of value*

int len = strlen(str);

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

if (isdigit(str[i]) == 0) {

++j;

**break**;

}

}

if (j == 0) **break**;

else printf("Need natural number!: ");

}

num = atoi(str); *//convert str to int*

return num;

}

int valid\_hex() {

char str[65];

int num;

while (1) {

int j = 0;

scanf("%s", str); *//input of value*

int len = strlen(str);

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

if (isxdigit(str[i]) == 0) {

++j;

**break**;

}

else if ((isalpha(str[i]) != 0) && (strlen(str) > 4)) {

printf("**\n**length of hex digit no more than 4**\n**");

++j;

**break**;

}

}

if (j == 0) **break**;

else printf("Need hex digit!: ");

}

num = (int)strtol(str, NULL, 16); *//convert str(hex) to int*

return num;

}

int main() {

char t; */\*тип пакета\*/*

char s; */\*ідентифікатор джерела\*/*

unsigned char l; */\*довжина пакета\*/*

unsigned int UnitStateWord; */\*слово стану\*/*

int choice;

printf("**\n**Sadchenko Nikita, student of KM-73.**\n**### Laboratory Work 1 ###**\n**");

do {

printf("**\n**Press '1' to form Program Status Word.**\n**"\

"Press '2' to get initial data. (Source ID, package type and length)**\n**"\

"Press '3' to clear console.**\n**"\

"Press '9' to get out program.**\n**");

choice = getch();

} while (!strchr("1239", choice));

switch (choice) {

case '1': {

printf("**\n**Enter package type (0-7): ");

while (1) {

int n = valid\_int();

if ((0 <= n) && (n <= 7)) {

t = (char) n; *// Assignment int to char (Присвоєння)*

**break**;

}

else printf("from '0' to '7': ");

}

printf("Enter Source ID (0-15): ");

while (1) {

int n = valid\_int();

if ((0 <= n) && (n <= 15)) {

s = (char) n;

**break**;

}

else printf("from '0' to '15': ");

}

printf("Enter package length (0-255): ");

while (1) {

int n = valid\_int();

if ((0 <= n) && (n <= 255)) {

l = (unsigned char) n;

**break**;

}

else printf("from '0' to '255': ");

}

*/\*формування упакованого коду\*/*

UnitStateWord = ((unsigned int) t & 7) << 13;

UnitStateWord |= ((unsigned int) s & 0xF) << 8;

UnitStateWord |= l & 0xFF;

*/\*слово стану пристрою\*/*

printf("**\n**Program Status Word = %04x**\n\n**", UnitStateWord);

**break**;

}

case '2': {

char ch[65];

printf("**\n**Enter Program Status Word**\n**");

printf("(hex digit from 0 to 0xFFFF): ");

int numb = valid\_hex();

sprintf (ch, "%x", numb); *//convert int(dex) to str(hex)*

UnitStateWord = (unsigned int)strtol(ch, NULL, 16); *//str(hex) to int(hex)*

*/\*виділення складових частин\*/*

t = (UnitStateWord >> 13) & 7;

s = (UnitStateWord >> 8) & 0xF;

l = UnitStateWord & 0xFF;

*/\*виведення результатів\*/*

putchar('**\n**');

printf("Package type = %d**\n**", t);

printf("Source ID = %d**\n**", s);

printf("Package length = %d**\n\n**", l);

**break**;

}

case '3': {

printf(" 7 ballov :)**\n**");

system("cls");

**break**;

}

case '9': exit(0); **break**;

default: **break**;

}

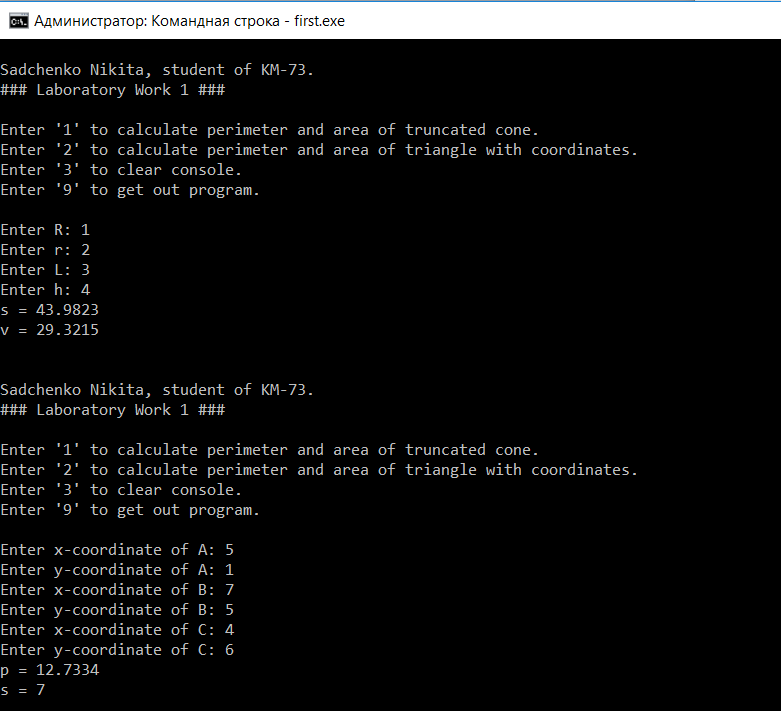
return main();

}

# 

ДОДАТОК В (Результат роботи програми)

## Перша програма



## 

## Друга програма

