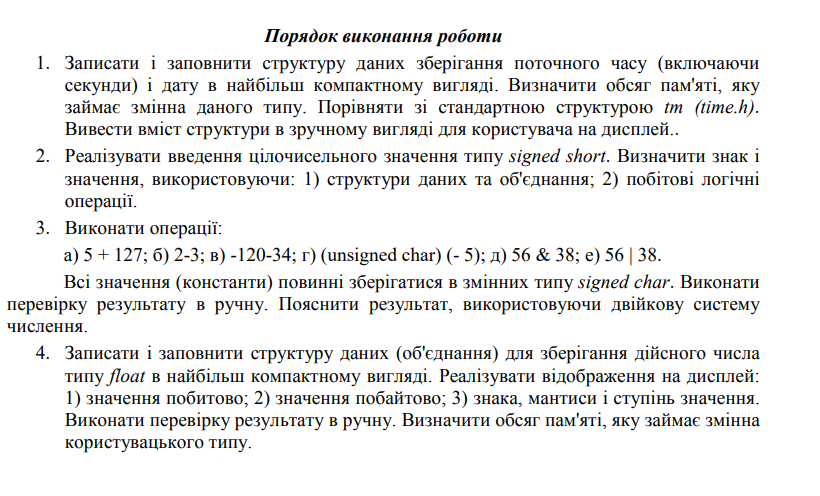
**Лабораторна робота № 1**

РОБОТА З БАЗОВИМИ ТИПАМИ ДАНИХ

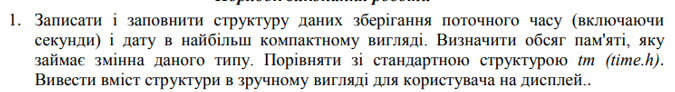
***Мета*** : отримати практичні навички по роботі з базовими типами даних (простими і складними типами даних).

**1.1 Хід роботи**

****

1.1.1

**Завдання 1**:



Лістинг:

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <windows.h>

#include <stdlib.h>

#include <time.h>

struct date {

unsigned short Year : 12;

unsigned short Month : 4;

unsigned short Day : 5;

unsigned short Hour : 5;

unsigned short Min : 6;

unsigned short Sec : 6;

};

enum dow {

Mon = 1,

Tue,

Wed,

Thu,

Fri,

Sat,

Sun

};

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

time\_t now = time(0);

struct date data;

int y, m, d, h, mm, s;

char k[20];

printf("Введіть номер дня тижня ");

dow doo;

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

switch (doo) {

case Mon:

strcpy\_s(k, "Понеділок");

break;

case Tue:

strcpy\_s(k, "Вівторок");

break;

case Wed:

strcpy\_s(k, "Середа");

break;

case Thu:

strcpy\_s(k, "Четвер");

break;

case Fri:

strcpy\_s(k, "П'ятниця");

break;

case Sat:

strcpy\_s(k, "Субота");

break;

case Sun:

strcpy\_s(k, "Неділя");

break;

default:break;

}

printf("Введіть рік:");

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

printf("Введіть місяць:");

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

printf("Введіть день:");

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

printf("Введіть годину:");

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

printf("Введіть хвилини:");

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

printf("Введіть секунди:");

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

data.Year = y;

data.Month = m;

data.Day = d;

data.Hour = h;

data.Min = mm;

data.Sec = s;

time\_t rawtime;

struct tm\* timeinfo;

time(&rawtime);

timeinfo = localtime(&rawtime);

printf("%s", asctime(timeinfo));

printf("%s, %d.%d.%d %d:%d:%d", k, data.Day, data.Month, data.Year, data.Hour, data.Min, data.Sec);

int n1, n2;

n1 = sizeof(tm);

n2 = sizeof(date);

printf("\nРозмір tm-%d розмір власної структури-%d", n1, n2);

return 0;

}

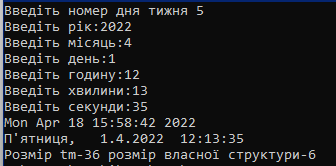
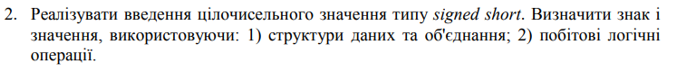


Рисунок 1.1 – Результат виконання завдання 1

1.1.2

**Завдання 2**:



Лістинг:

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <string.h>

#include <windows.h>

#include <math.h>

#include <conio.h>

union binary {

signed short number\_binary;

struct {

unsigned char zero : 1;

unsigned char one : 1;

unsigned char two : 1;

unsigned char three : 1;

unsigned char four : 1;

unsigned char five : 1;

unsigned char six : 1;

unsigned char seven : 1;

unsigned char eight : 1;

unsigned char nine : 1;

unsigned char ten : 1;

unsigned char eleven : 1;

unsigned char twelve : 1;

unsigned char thirteen : 1;

unsigned char fourteen : 1;

unsigned char fiveteen : 1;

}bytes;

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

signed short num;

binary binary;

do {

printf("\n\nДля закінчення програми введіть 0.\nВведіть ціле значення = ");

scanf("%d", &num);

binary.number\_binary = num;

if (binary.bytes.fiveteen == 1) {

printf("\nЧисло %d є від'ємним, бо знаковий біт = %d.", binary.number\_binary, binary.bytes.fiveteen);

}

else {

printf("\nЧисло %d є додатнім, бо знаковий біт = %d.", binary.number\_binary, binary.bytes.fiveteen);

}

} while (num != 0);

return 0;

}

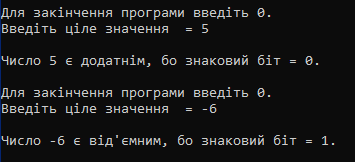
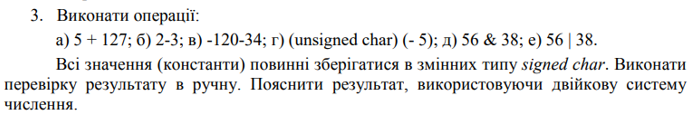


Рисунок 1.2 – Результат виконання завдання 2

1.1.2

**Завдання 3**:

Лістинг:



#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <math.h>

#include <Windows.h>

#include <string.h>

#include <conio.h>

#include <time.h>

struct ch {

signed char a = 5;

signed char b = 127;

signed char c = 2;

signed char d = -3;

signed char f = -120;

signed char e = -34;

signed char g = -5;

signed char l = 56;

signed char k = 38;

signed char rez;

}x;

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

x.rez = x.a + x.b;

printf("5+127=%d", x.rez);

printf("\nВиходить за межі значень типу signed char");

x.rez = 0;

x.rez = x.c + x.d;

printf("\n\n2-3=%d", x.rez);

printf("\nНе виходить за межі значень типу signed char");

x.rez = 0;

x.rez = x.f + x.e;

printf("\n\n-120-34=%d", x.rez);

printf("\nВиходить за межі значень типу signed char");

x.rez = 0;

x.rez = (unsigned char)(x.g);

printf("\n\nunsigned char -5=%d", x.rez);

printf("\nНічого не відбувається");

x.rez = 0;

x.rez = x.l & x.k;

printf("\n\n56 & 38=%d", x.rez);

printf("\nЛогічна дія AND");

x.rez = 0;

x.rez = x.l | x.k;

printf("\n\n56 | 38=%d", x.rez);

printf("\nЛогічна дія OR\n");

return 0;

}

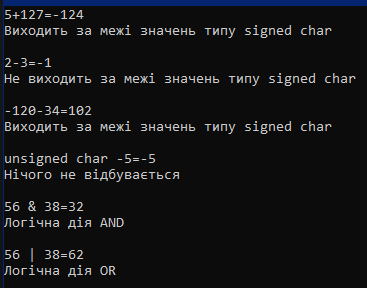
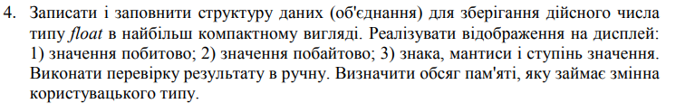


Рисунок 1.3 – Результат виконання завдання 3

1.1.4

**Завдання 4**:



Лістинг:

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <math.h>

#include <Windows.h>

#include <string.h>

#include <conio.h>

#include <time.h>

union radiance {

float number;

struct {

unsigned char zero : 1;

unsigned char one : 1;

unsigned char two : 1;

unsigned char three : 1;

unsigned char four : 1;

unsigned char five : 1;

unsigned char six : 1;

unsigned char seven : 1;

unsigned char eight : 1;

unsigned char nine : 1;

unsigned char ten : 1;

unsigned char eleven : 1;

unsigned char twelve : 1;

unsigned char thirteen : 1;

unsigned char fourteen : 1;

unsigned char fiveteen : 1;

unsigned char sixteen : 1;

unsigned char seventeen : 1;

unsigned char eighteen : 1;

unsigned char nineteen : 1;

unsigned char twenty : 1;

unsigned char twenty\_one : 1;

unsigned char twenty\_two : 1;

unsigned char twenty\_three : 1;

unsigned char twenty\_four : 1;

unsigned char twenty\_five : 1;

unsigned char twenty\_six : 1;

unsigned char twenty\_seven : 1;

unsigned char twenty\_eight : 1;

unsigned char twenty\_nine : 1;

unsigned char thirty : 1;

unsigned char thirty\_one : 1;

}bytes;

}radiance;

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

printf("Введіть змінну = "); scanf("%f", &radiance.number);

printf("\nЗначення побитово:\n");

printf("|%d|%d %d %d %d %d %d %d %d|%d %d %d %d %d %d %d %d %d %d %d %d %d %d %d % d % d % d % d % d % d % d % d | \n", radiance.bytes.thirty\_one, radiance.bytes.thirty,radiance.bytes.twenty\_nine, radiance.bytes.twenty\_eight,radiance.bytes.twenty\_seven, radiance.bytes.twenty\_six,radiance.bytes.twenty\_five, radiance.bytes.twenty\_four,radiance.bytes.twenty\_three, radiance.bytes.twenty\_two,radiance.bytes.twenty\_one, radiance.bytes.twenty, radiance.bytes.nineteen,radiance.bytes.eighteen, radiance.bytes.seventeen, radiance.bytes.sixteen,radiance.bytes.fiveteen, radiance.bytes.fourteen, radiance.bytes.thirteen,radiance.bytes.twelve, radiance.bytes.eleven, radiance.bytes.ten,radiance.bytes.nine, radiance.bytes.eight, radiance.bytes.seven,radiance.bytes.six, radiance.bytes.five, radiance.bytes.four,radiance.bytes.three, radiance.bytes.two, radiance.bytes.one,radiance.bytes.zero);

printf(" | | |\n");

printf("знак ступінь мантиса\n");

printf("\nЗначення побайтово:\n");

printf("|%d %d %d %d %d %d %d %d|%d %d %d %d %d %d %d %d|%d %d %d %d %d %d %d % d | % d % d % d % d % d % d % d % d | \n", radiance.bytes.thirty\_one, radiance.bytes.thirty,radiance.bytes.twenty\_nine, radiance.bytes.twenty\_eight,radiance.bytes.twenty\_seven, radiance.bytes.twenty\_six,radiance.bytes.twenty\_five, radiance.bytes.twenty\_four,radiance.bytes.twenty\_three, radiance.bytes.twenty\_two,radiance.bytes.twenty\_one, radiance.bytes.twenty, radiance.bytes.nineteen,radiance.bytes.eighteen, radiance.bytes.seventeen, radiance.bytes.sixteen,radiance.bytes.fiveteen, radiance.bytes.fourteen, radiance.bytes.thirteen,radiance.bytes.twelve, radiance.bytes.eleven, radiance.bytes.ten,radiance.bytes.nine, radiance.bytes.eight, radiance.bytes.seven,radiance.bytes.six, radiance.bytes.five, radiance.bytes.four,radiance.bytes.three, radiance.bytes.two, radiance.bytes.one,radiance.bytes.zero);

float s;

printf("\nОбяг пам'яті змінної = %d;\nОбсяг пам'яті union = % d.", sizeof(s), sizeof(radiance));

return 0;

}

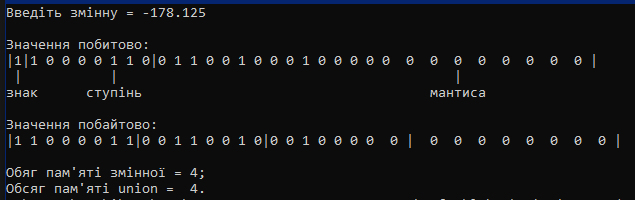


Рисунок 1.4 – Результат виконання завдання 4

***Висновки:*** я отримав практичні навички по роботі з базовими типами даних (простими і складними типами даних) та навчився складати за ними програми.