**1 ЦЕЛЬ И ЗАДАЧИ ЛАБОРАТОРНОЙ РАБОТЫ**

**1.1** Цель лабораторной работы – освоить механизм структур языка С, научиться работать с массивами пользовательских типов данных.

**1.2** Задача лабораторной работы – написать программу для работы с массивом структур.

**2 РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ**

**2.1 Условие:**

- Реализовать функцию для создания массива структур.

- Реализовать функцию для вывода массива структур на экран.

- Реализовать функцию для инициализации экземпляра структуры.

- Реализовать функции сортировки массива по каждому полю структуры.

- Реализовать меню, для проверки функционала.

- Реализовать функцию для удаления структуры из массива

**2.2** На рисунке.1 – 2 приведена блок-схема алгоритма.

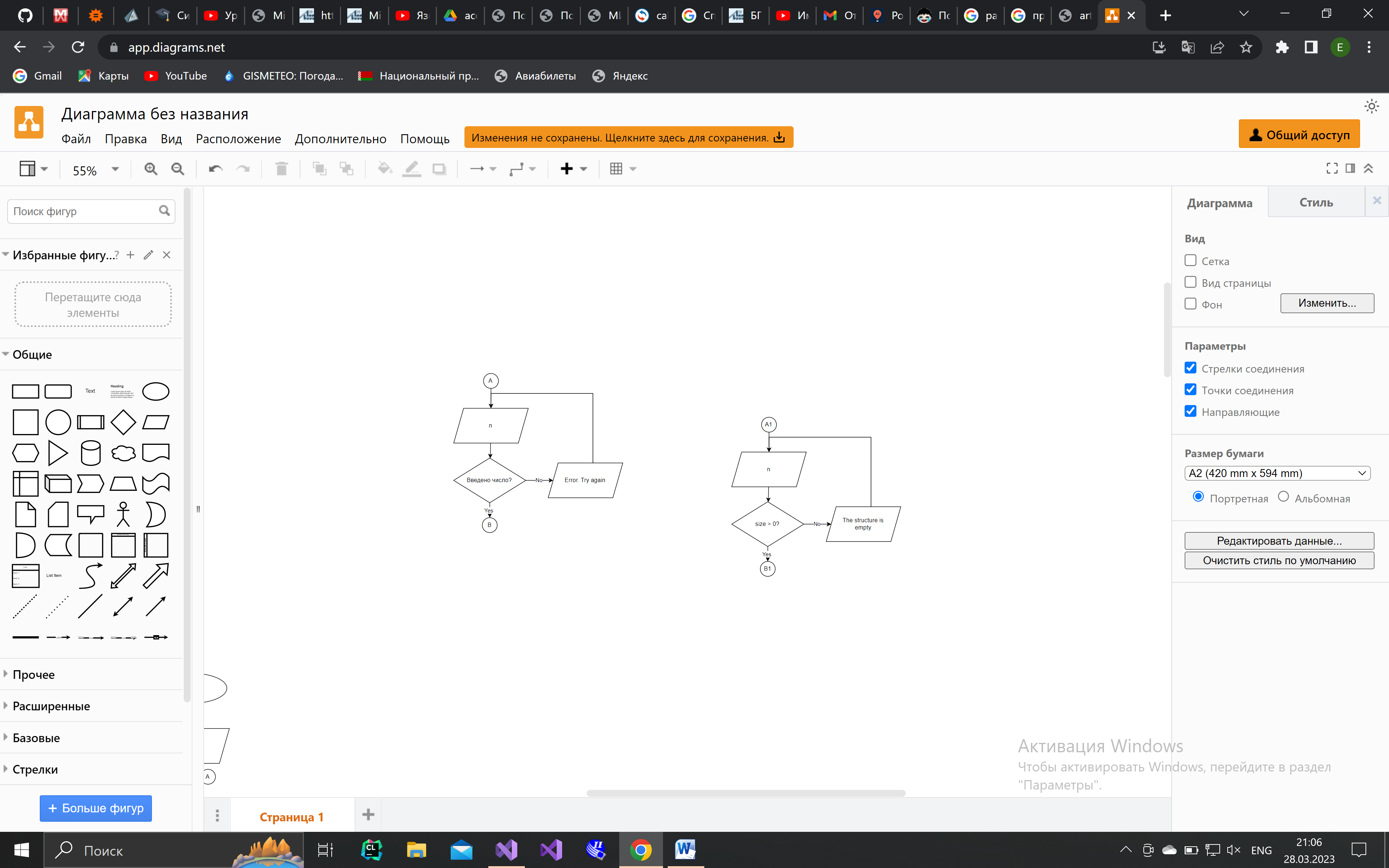


Рисунок 1

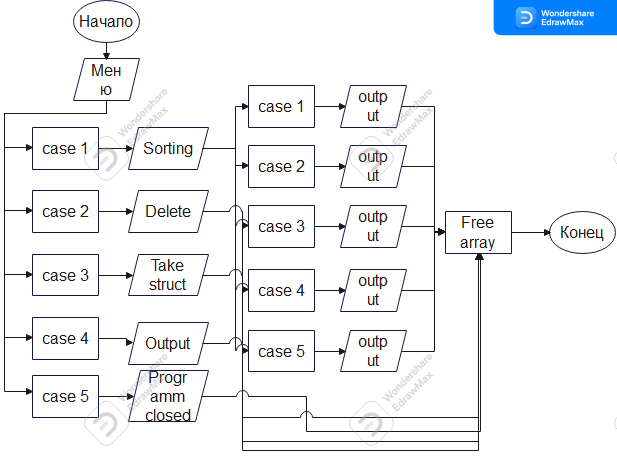


Рисунок 2

**2.3** Листинг компьютерной программы:

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

typedef struct {

char name[100];

int speed;

int armorp;

int year;

float caliber;

} Tanks;

Tanks\* create\_array(int n) {

Tanks\* array = malloc(n \* sizeof(Tanks));

return array;

}

void print\_array(Tanks\* array, int n) {

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

printf("Name: %s\nSpeed: %d\nArmor penetration: %d\nYear: %d\nCaliber: %f\n\n", array[i].name, array[i].speed, array[i].armorp, array[i].year, array[i].caliber);

}

}

void init\_struct(Tanks\* s) {

printf("Enter tank name: ");

fgets(s->name, 100, stdin);

s->name[strcspn(s->name, "\n")] = 0;

printf("Enter tank speed: ");

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

printf("Enter the armor penetration of the tank: ");

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

printf("Enter the year of creation: ");

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

printf("Enter caliber: ");

scanf\_s(" %f", &s->caliber);

}

void sort\_array(Tanks\* array, int n, int field) {

for (int i = 0; i < n - 1; i++) {

for (int j = 0; j < n - i - 1; j++) {

int swap = 0;

switch (field) {

case 1:

if (strcmp(array[j].name, array[j + 1].name) > 0)

swap = 1;

break;

case 2:

if (array[j].speed > array[j + 1].speed)

swap = 1;

break;

case 3:

if (array[j].armorp > array[j + 1].armorp)

swap = 1;

break;

case 4:

if (array[j].year > array[j + 1].year)

swap = 1;

break;

case 5:

if (array[j].caliber > array[j + 1].caliber)

swap = 1;

break;

}

if (swap) {

Tanks temp = array[j];

array[j] = array[j + 1];

array[j + 1] = temp;

}

}

}

}

void delete\_struct(Tanks\*\* array, int\* n, int index) {

memmove(\*array + index, \*array + index + 1, (\*n - index - 1) \* sizeof(Tanks));

\*n -= 1;

\*array = realloc(\*array, \*n \* sizeof(Tanks));

}

int main() {

int n = 0;

Tanks\* array = create\_array(n);

int choice;

do {

printf("Vehicle List:\n");

printf("1. Sort vehicle\n");

printf("2. Delete vehicle\n");

printf("3. Add vehicle\n");

printf("4. Print vehicle list\n");

printf("5. Exit\n");

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

int c;

while ((c = getchar()) != '\n' && c != EOF) {}

switch (choice) {

case 1:

printf("Enter field to sort (1 - name; 2 - speed; 3 - armor penetration; 4 - year; 5 - caliber): ");

int field;

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

sort\_array(array, n, field);

break;

case 2:

printf("Enter index to delete: ");

int index;

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

delete\_struct(&array, &n, index);

break;

case 3:

n += 1;

array = realloc(array, n \* sizeof(Tanks));

init\_struct(&array[n - 1]);

break;

case 4:

print\_array(array, n);

}

}

while (choice != 5);

free(array);

return 0;

}

**2.4** Результат выполнения компьютерной программы на экран в виде «скриншота» изображения на мониторе представлены на рисунках 4 – 8:

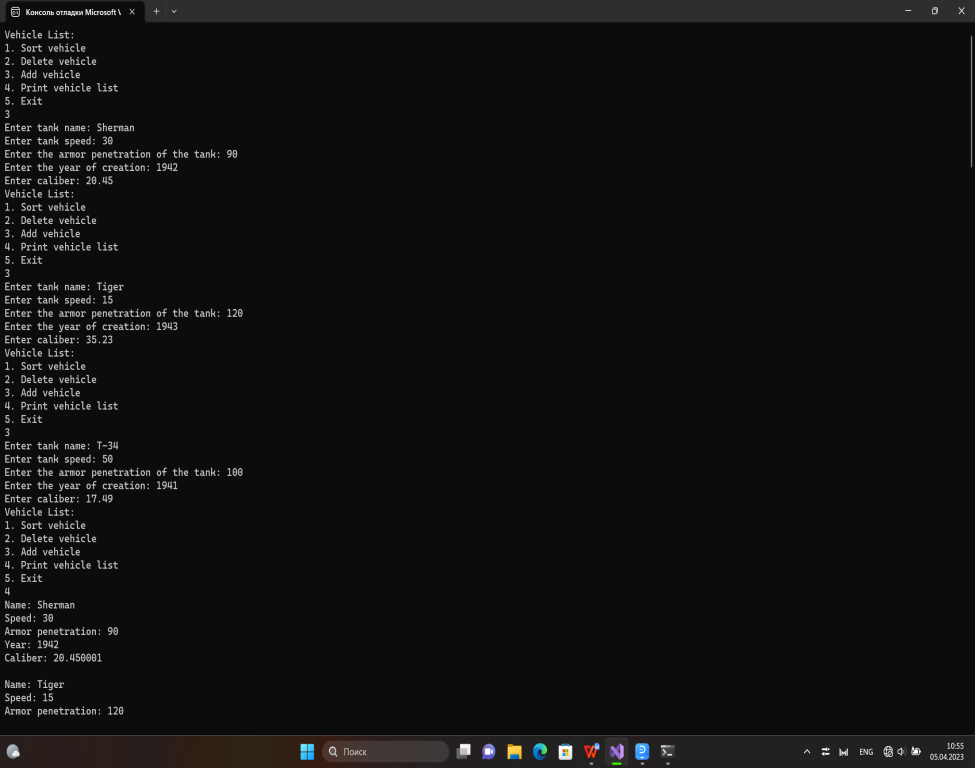


Рисунок 4

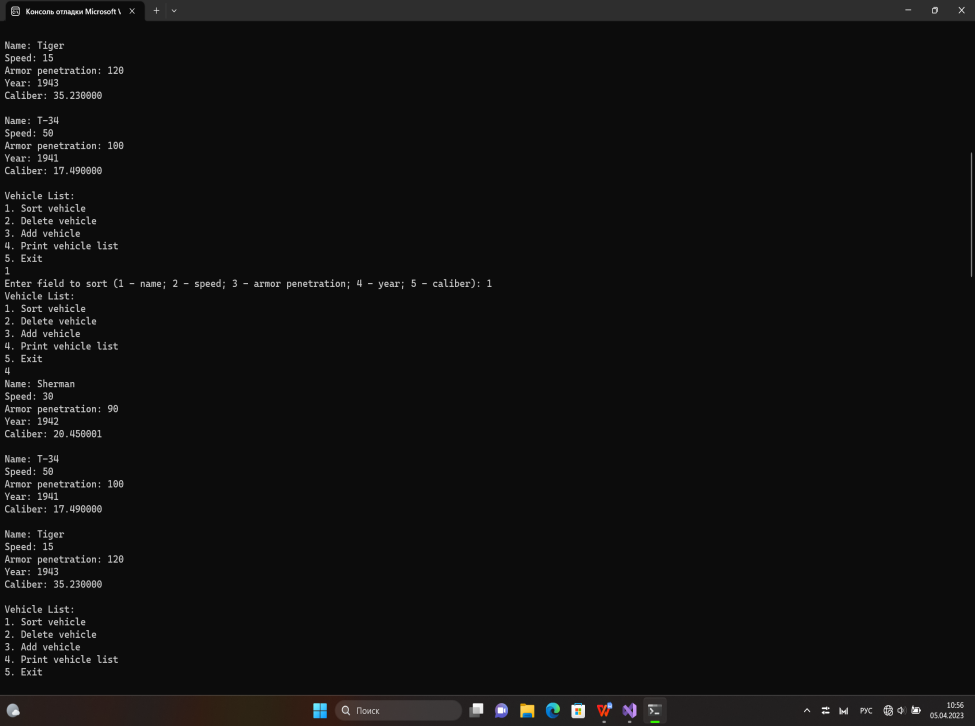


Рисунок 5

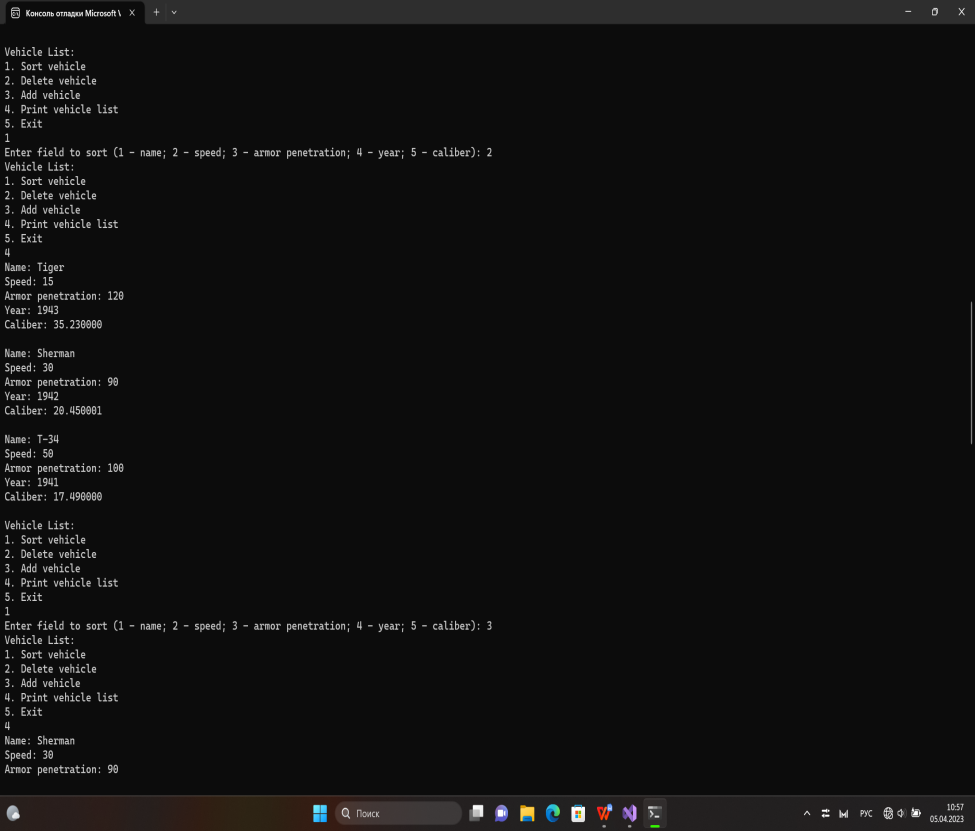


Рисунок 6

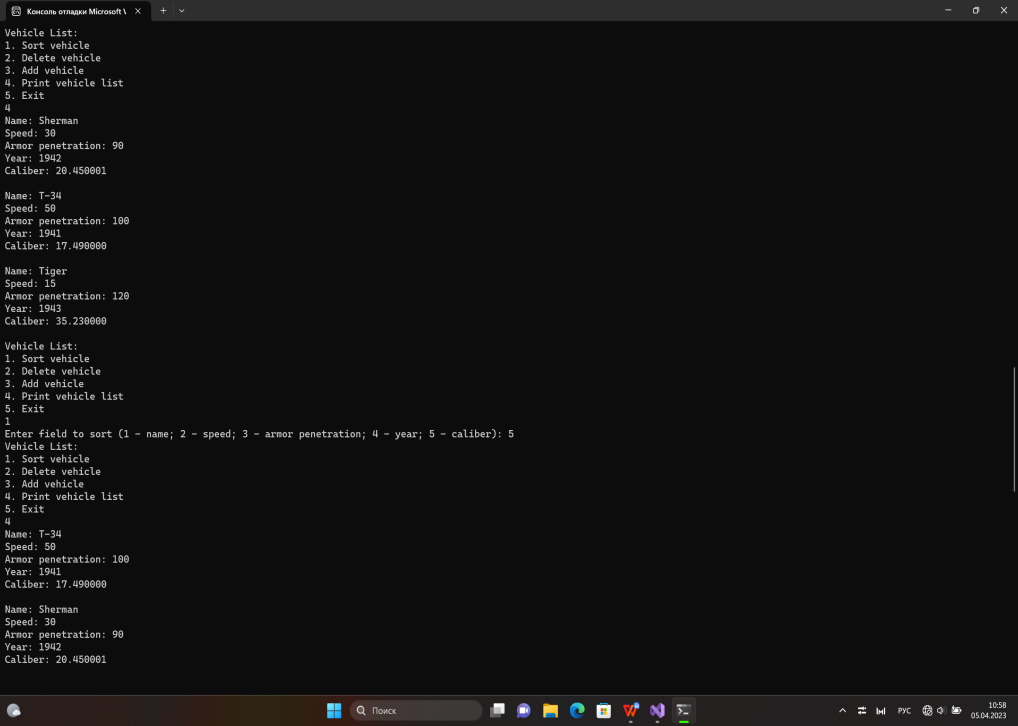


Рисунок 7

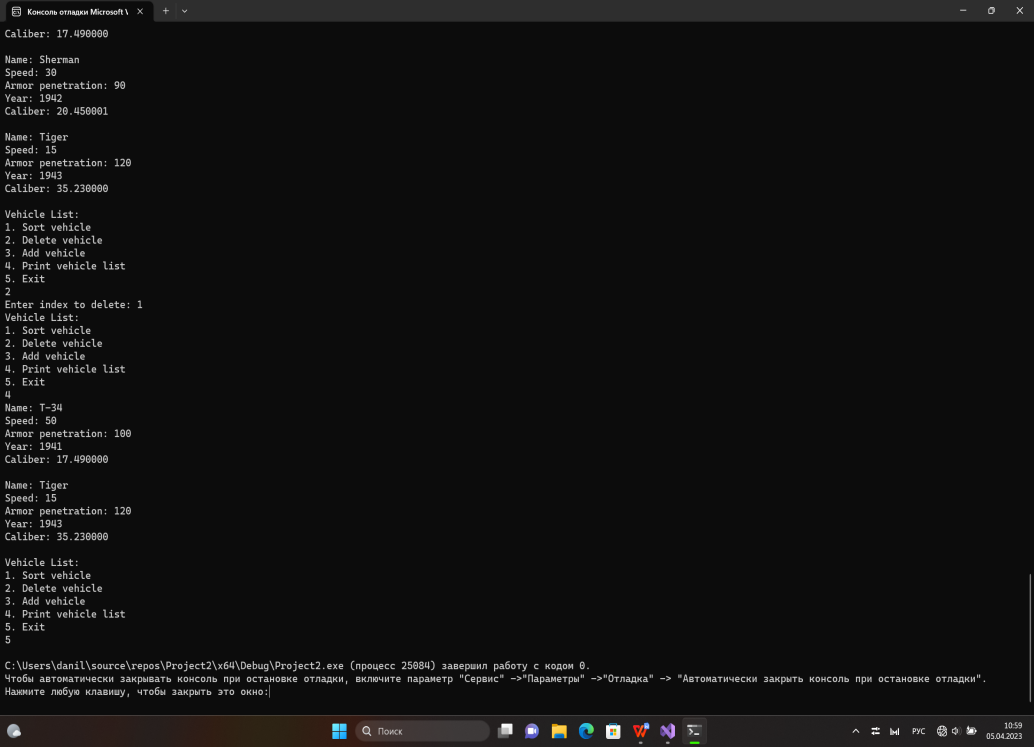


Рисунок 8