|  |  |
| --- | --- |
| Изображение выглядит как текст, небо, внешний  Автоматически созданное описание | |
| Національний технічний університет України  «Київський політехнічний інститут ім. Ігоря Сікорського»    Інститут прикладного системного аналізу  Кафедра системного проектування | |
| **Звіт** | |
| **про виконання лабораторної роботи №4** | |
| **з дисципліни «Алгоритмізація та програмування»** | |
| Виконав:  Студент гр. ДА-12  Панасенко О. А.  Прийняв:  к.т.н., доцент Безносик О. Ю. |  |
| Київ - 2021 | |

**Мета роботи**

Оволодіння навичками роботи із структурованими типами даних – масивами, особливостями введення, обробки і виведення. Обробка одновимірних масивів, матриць.

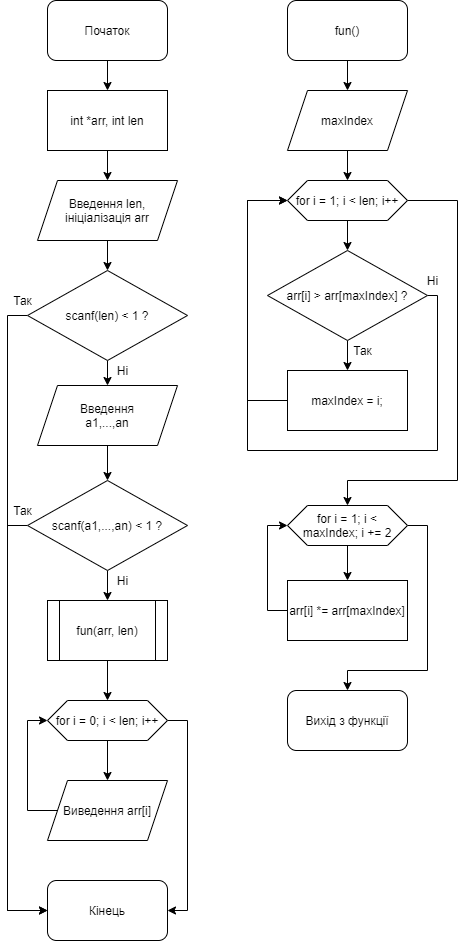
**Завдання 21**

Дані цілі числа а1,...,an. Всі члени послідовності з парними номерами, попередні першому по порядку члену із значенням max (а1,...,an), домножить на max (а1,...,an).

**1. Аналіз умови задачі.**

Оскільки я маю справу з числовою послідовністю, створю масив цілих чисел (тип даних *int*) з довжиною, що буде вводитись з клавіатури. Послідовність також буде задаватись з клавіатури. Аби виконати завдання треба знайти індекс найбільшого елементу масива та домножити усі попередні йому парні елементи на це значення.

Блок-схему алгоритму можна побачити на рисунку 1.



*Рисунок 1. Блок-схема програми*

**2. Код програми та результат виконання програми:**

|  |
| --- |
| #include <stdio.h> #include <malloc.h> #include <windows.h>  void fun(int \*arr, int len) {  // get max  int maxIndex = arr[0];  for (int i = 1; i < len; i++)  if (arr[i] > arr[maxIndex]) maxIndex = i;  // multiply  for (int i = 1; i < maxIndex; i += 2)  arr[i] \*= arr[maxIndex]; }  int main() {  SetConsoleOutputCP(CP\_UTF8);  int \*arr, len;  printf("Введите длину массива:\n");  if (scanf("%d", &len) < 1) {  printf("Указан неверный тип данных. Нужно число\n");  return -1;  }  arr = malloc(len \* sizeof(int));  printf("Введите %d элементов массива через пробел:\n", len);  for (int i = 0; i < len; i++) {  if (scanf("%d", &arr[i]) < 1) return -2;  }  fun(arr, len);  for (int i = 0; i < len; i++) printf("%d ", arr[i]);  return 0; } |

На рисунку нижче зображено результат виконання програми.

Изображение выглядит как текст

Автоматически созданное описание

*Рисунок 2. Результат виконання програми*

**Висновки**

У процесі виконання *лабораторної роботи №4* я розробив алгоритм для обробки елементів заданої послідовності, знаходження її максимального значення і реалізував цей алгоритм на *мові програмування C*. Я створив окрему функцію, яка знаходить найбільше значення послідовності. Алгоритм реалізовано через 2 цикли for, хоча можна було скористатися одним, але більш громіздким. Всі використані змінні мають тип – *int*, оскільки надана послідовність є цілочисленною. Щоб переконатися що користувач ввів число, порівнюю функцію *scanf* і 1. Також я навчився виділяти динамічну пам’ять під масив за допомогою функції *malloc* з однойменної бібліотеки.