

ЛАБОРАТОРНА РОБОТА № 20

Рекурсія

Мета: Формування навиків роботи із функціями. Вивчення методів використання алгоритмів і програм з рекурсією в мові Сі.

Хід роботи:

Завдання 1.

3	За допомогою рекурсивної функції здійснити пошук мінімального елемента одновимірного масиву
---	---

Рис. 1. Завдання для написання першої програми

Лістинг програми:

```
#include <stdio.h>
#include <windows.h>
#include <time.h>
#include <stdlib.h>
#define n 10

int minarr(int *p, int, int);
int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    srand(time(NULL));
    int arr[n];
    int* p = &arr[0], i = 0;
    printf("Arr = { ");
    for (int i = 0; i < n; i++) {
        arr[i] = rand() % 21 - 10;
        printf("%d", arr[i]);
        if (i != n - 1) printf(", ");
    }
    int min = *p;
    printf(" }\n");
    printf("\nМінімальний елемент масиву = %d\n", minarr(p, i, min));
    return 0;
}

int minarr(int *p, int i, int min) {
    if (*(p + i) <= min) min = *(p + i);
    i++;
    if (i == n)
        return min;
    return minarr(p, i, min);
}
```

					ДУ «Житомирська політехніка».22.122.13.000 – Лр20			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Черниш М.			Звіт з лабораторної роботи		Лім.	Арк.
Перевір.		Терещук С.О.						1
Керівник								4
Н. контр.							ФІКТ Гр. КН-22-3[2]	
Зав. каф.								

Результат виконання програми:

```

Microsoft Visual Studio Debug Console
Arr = { -1, -3, -3, -6, 3, 3, -7, -10, -4, -2 }
Мінімальний елемент масиву = -10
D:\КН-22-3[2]\Основи програмування\Лабораторні роботи

```

Рис. 2. Результат виконання першої програми

Завдання 2. Вирішити задачу двома способами - із застосуванням рекурсії і без неї.

3	Знайти максимальний елемент масиву a_1, \dots, a_n , використовувати метод ділення навпіл $\max(a_1, \dots, a_n) = \max(\max(a_1, \dots, a_{n/2}), \max(a_{n/2+1}, \dots, a_n))$.
---	--

Рис. 3. Завдання для написання другої програми

Лістинг програми:

```

#include <stdio.h>
#include <windows.h>
#include <time.h>
#include <stdlib.h>

int maxxarray(int arr[], int i, int n, int max, int max1) {
    if (i < (n - 1) / 2) {
        if (arr[i] >= max) max = arr[i];
        i++;
        return maxxarray(arr, i, n, max, max1);
    }
    if (i <= n) {
        if (arr[i] >= max1) max1 = arr[i];
        i++;
        return maxxarray(arr, i, n, max, max1);
    }
    int maxxarr;
    if (max >= max1) maxxarr = max;
    else maxxarr = max1;
    return maxxarr;
}

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    srand(time(NULL));
    const int n = 10;
    int arr[n], i = 0;
    printf("Arr = { ");
    for (int i = 0; i < n; i++) {
        arr[i] = rand() % 21 - 10;
        printf("%d", arr[i]);
        if (i != n - 1) printf(", ");
    }
    printf(" }\n");
    int max = arr[0], max1 = arr[n / 2];

```

```

printf("\nМаксимальний елемент масиву(через рекурсію) = %d\n", maxarray(arr, i, n,
max, max1));
int max2 = arr[0], maxa;
for (int i = 0; i < (n - 1) / 2; i++)
    if (arr[i] >= max2) max2 = arr[i];
int max3 = arr[n / 2];
for (int i = n / 2; i < n; i++)
    if (arr[i] >= max3) max3 = arr[i];
if (max2 >= max3) maxa = max2;
else maxa = max3;
printf("\nМаксимальний елемент масиву(без рекурсії) = %d\n", maxa);
return 0;
}

```

Результат виконання програми:

```

Microsoft Visual Studio Debug Console
Arr = { 5, 0, 0, 6, 4, -1, 8, -10, -1, 9 }
Максимальний елемент масиву(через рекурсію) = 9
Максимальний елемент масиву(без рекурсії) = 9
D:\КН-22-3[2]\Основи програмування\Лабораторні роботи

```

Рис. 4. Результат виконання другої програми

Завдання на самостійну роботу:

1	Перевірити, чи є задане число паліндромом.
3	Підрахувати $y=x^N$ за наступним алгоритмом $y=(x^{N/2})^2$, якщо N парне, $y=x*x^{N-1}$, якщо N непарне.

Рис. 5. Завдання для написання програм самостійної роботи

Лістинг програми(1):

```

#include <stdio.h>
#include <windows.h>

int palindrom(int n, int m, int a) {
    a *= 10;
    a += n % 10;
    n /= 10;
    if (n == 0) {
        if (m == a) printf("\nЧисло %d є паліндромом\n", m);
        else printf("\nЧисло %d не є паліндромом\n", m);
    }
    else return palindrom(n, m, a);
}

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    int n, a = 0;
    printf("Введіть число: "); scanf_s("%d", &n);
}

```

```

int m = n;
palindrom(n, m, a);
return 0;
}

```

Результат виконання програми:

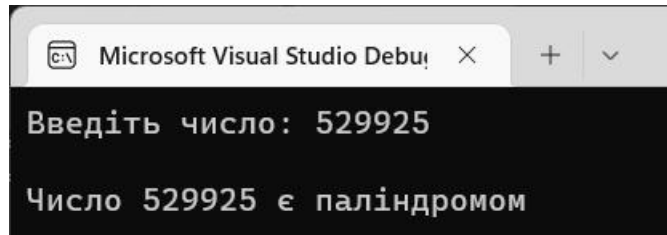


Рис. 6. Результат виконання першої програми самостійної роботи

Лістинг програми(2):

```

#include <stdio.h>
#include <windows.h>

void power(int x, int n, int y, int i) {
    if (n % 2 == 0) {
        if (i < n / 2) {
            y *= x;
            return power(x, n, y, i + 1);
        }
        y *= y;
    }
    else {
        if (i < n - 1) {
            y *= x;
            return power(x, n, y, i + 1);
        }
        y *= x;
    }
    printf("\nВідповідь: y = %d^%d = %d\n", x, n, y);
}

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    int y = 1, x, n, i = 0;
    printf("Введіть число: "); scanf_s("%d", &x);
    printf("Введіть степінь числа: "); scanf_s("%d", &n);
    power(x, n, y, i);
    return 0;
}

```

Результат виконання програми:

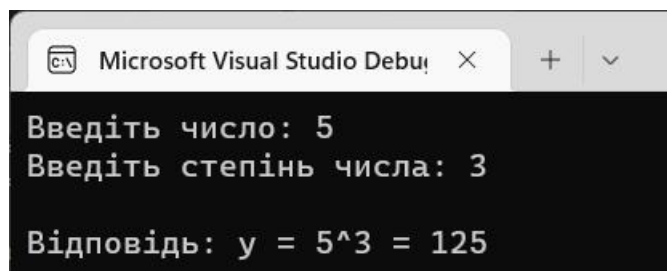


Рис. 7. Результат виконання другої програми самостійної роботи

		Черниш М.			ДУ «Житомирська політехніка».22.122.13.000 – Лр20	Арк.
		Герецьук С.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		4