НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. І. СІКОРСЬКОГО»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

КАФЕДРА СИСТЕМНОГО ПРОГРАМУВАННЯ ТА СПЕЦІАЛІЗОВАНИХ КОМП’ЮТЕРНИХ СИСТЕМ

**Лабораторна робота №6  
з дисципліни «Системне програмне забезпечення»**

**Варіант 9**

Виконав  
студент 4-го курсу  
групи КВ-41  
Курач Віктор

Київ – 2017

**Постановка задачі**

1. Проаналізувати програму, виміряти час її виконання, визначити, де є простір для поліпшення.
2. Виходячи з описаних у теорії механізмів організації пам'яті внести необхідні зміни в структуру коду.
3. Виміряти новий час виконання, за наявності очевидних поліпшень, переходити до п.4, якщо немає - до п.1
4. Зафіксувати нові показники і пояснити, за рахунок чого було досягнуто прискорення.

**Завдання для даного варіанту**

#include<stdio.h>

void new\_func1(void);

int func1(int a, int b){

int res = 0;

for(int i = 0; i < 10; i++) {

int res = func2(a, b);

if(res > 0) return res;

}

return;

}

int func2(int a, int b) {

int res = 0

for(int i = 0; i < 10; i++) {

sleep(1);

if(i>8) res = resultOfSum(a, b);

if(res > 0) return res;

}

return res;

}

int resultOfSum(int a, int b) {

return a+b;

}

int main(void) {

func1(5, 10);

return 0;

}

**Лістинг програми**

#include <stdio.h>

#include <windows.h>

#include <time.h>

#define N 1000000000

int func1(int a, int b);

int func2(int a, int b);

int resultOfSum(int a, int b);

int func1(int a, int b) {

int res = 0;

long i;

for (i = 0; i < N; i++) {

res = func2(a, b);

if (res > 0) return res;

}

return res;

}

int func2(int a, int b) {

int res = 0;

long i;

for (i = 0; i < N; i++) {

//Sleep(1);

if (i > N-2) res = resultOfSum(a, b);

if (res > 0) return res;

}

return res;

}

int resultOfSum(int a, int b) {

return a + b;

}

void main() {

time\_t start, end;

int res;

start = time(0);

res = func1(5, 10);

end = time(0);

printf("%.20f\n", difftime(end, start));

}

**Аналіз та висновки**

Час виконання програми – близько **4 с**. При цьому, при заданих вхідних даних, переважна більшість часу витрачається на виконання функції *func2*. Оскільки (при заданих вхідних даних) її результат додатній, цикл у функції *func1* виконує лише одну ітерацію. Тому, задля зменшення витрат пам’яті, цикл можна було би прибрати, але якщо для заданих вхідних даних результат обчислень не зміниться, не можна стверджувати, що це справедливо для інших вхідних даних (наприклад, якщо *a* і *b* будуть від’ємні).

Єдина можлива оптимізація заданої програми – заміна виклику функції *resultOfSum* оператором «+», оскільки функція викликається лише один раз.