Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики і обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №6

з курсу: «Організація обчислювальних процесів»

Виконала:

студент групи ІВ-71

Молчанова В.С.

Залікова книжка №7110

Перевірив:

Сімоненко В.П.

Київ 2020 р.

**Тема:** Профілювання програм

**Мета:** Оптимізувати програму із використанням інструментів профілювання

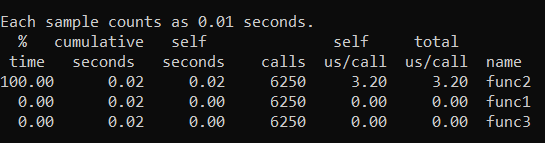
**Варіант:** (7110 mod 15) + 1 = 11

**Виконання роботи:**

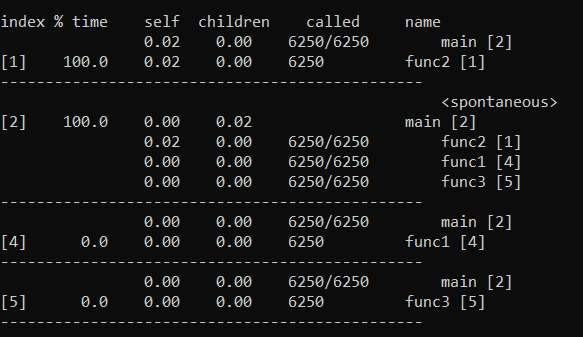
Лістинг початкової програми:

#include **<stdio.h>  
  
void** new\_func1();  
  
**int** func1(**int** a) {  
 **for**(**int** i = 0; i < 300; i++) {  
 **if**(i>a) **return** i;  
 }  
 **return** a;  
}  
  
**void** func2() {  
 **for**(**int** i = 0; i < 1300; i++) {  
 }  
}  
  
**int** func3(**int** b) {  
 **for**(**int** i = 500; i > 3; i--) {  
 **if**(i<b) **return**;  
 }  
 **return** b;  
}  
  
**int** main() {  
 printf(**"\n Inside main()\n"**);  
  
 **for**(**int** i = 0; i < 25; i++) {  
 **for**(**int** j=250; j > 0; j--) {  
 printf(**"\n Inside for\n"**);  
 func1(i);  
 func2();  
 func3(i);  
 }  
 }  
  
 **return** 0;  
}

Аналіз:



Як бачимо, хоч час виконання і не дуже критичний (0.02 с), однак у функцій дуже велике число викликів. Поглянемо на граф викликів щоб розібратись більш детально:



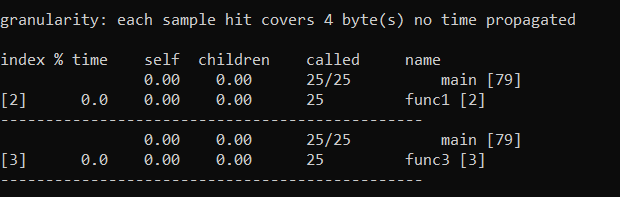
Всі три функції викликаються в main(), кожна по 6250 разів. Якщо подивитись на самі функції, то видно, що func2() не виконує жодної корисної роботи, а лише ітерується по циклам без мети. Можемо її прибрати з main(), а великі цикли в func1() та func3() замінити на логічні вирази, які їм еквівалентні. Результат виконання програми не зміниться, а 3 цикли не будуть займати ресурси без жодної користі:

Лістинг отриманої програми:

#include **<stdio.h>  
  
int** func1(**int** a) {  
 **if** (a < 30) {  
 **return** a + 1;  
 } **else** {  
 **return** a;  
 }  
}  
  
**int** func3(**int** b) {  
 **if** (b > 3 && b < 50) {  
 **return** b - 1;  
 } **else** {  
 **return** b;  
 }  
}  
  
**int** main() {  
 printf(**"\n Inside main()\n"**);  
  
 **for**(**int** i = 0; i < 25; i++) {  
 printf(**"\n Inside for\n"**);  
 func1(i);  
 func3(i);  
 }  
  
 **return** 0;  
}

Аналіз:

Одразу поглянемо на граф викликів для наочності:



Час виконання зменшився до такого, що gprof його навіть не фіксує.