

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ  
“ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**

**Лабораторна робота №9**

з дисципліни

«Операційні системи»

**Виконала:**

Студентка групи КН-  
214

Олескевич Софія

**Викладач:**

Кривенчук Ю.П.

Львів 2020

## Лабораторна робота №9

**Тема.** Виконання задачі в декількох потоках в ОС Linux

**Мета.** Навчитись реалізовувати розпаралелювання алгоритмів за допомогою багатопоточності в ОС Linux з використанням пакету функцій pthread.

**Завдання.**

1. Реалізувати заданий алгоритм в окремому потоці з використанням функцій з пакету pthread (завдання взяти з лабораторної роботи № 3).
2. Виконати розпаралелювання заданого алгоритму на 2, 4, 8 потоків.
3. Порівняти результати виконання програми під ОС Windows та Linux.
4. Результати виконання роботи відобразити у звіті.

### Варіант 19

**CODE:**

```
#include <iostream>

#include <stdio.h>

#include <sys/types.h>

#include <sys/stat.h>

#include <dirent.h>

#include <unistd.h>

#include <vector>

#include <set>

#include <filesystem>

#include <time.h>

#include <pthread.h>

using namespace std;

int cnt=0, files=0, catalogs=0, per_thread, size0, num, pr;

vector < pair<string,string> > names;

vector <long> si;
```

```
vector <string> res;
```

```
void *sort(void* i) {  
    long n;  
    int l = n * per_thread;  
    int r = (n + 1) * per_thread;  
    if(n == per_thread)  
        r = names.size();  
    cout << "Thread with ID " << pthread_self() << " has started!" << endl;  
    cout << "Interval limits from " << l + 1 << " " << r << endl;  
    for(int i=l; i<r; i++)  
    {  
        if(si[i]==size0)  
        {  
            res.push_back(names[i].first);  
        }  
        if(i==num)  
            nice(pr);  
  
        sleep(6);  
    }  
    cout << "Thread with ID " << pthread_self() << " has ended!" << endl;  
    return NULL;  
}
```

```
void find(string s, string no){  
    DIR *dir1;  
    struct dirent *info1;  
    struct stat a;
```

```

dir1 = opendir(s.c_str());
if(dir1 && no!="." && no!="..")
{
    names.push_back({s,"catalog"});
    catalogs++;
    while ((info1 = readdir(dir1)) != NULL) {
        string s1(info1->d_name);
        if (s1 != "." && s1 != "..") {
            stat(s.c_str(),&a);
            si.push_back(info1->d_reclen);
            cnt++;
            string s2 = s + "/";
            string s1(info1->d_name);
            s2 += s1;
            find(s2, s1);
            cnt--;
        }
    }
}
else { names.push_back({s,"file"});}
closedir(dir1);
}

int main()
{
    find("/home","");
    cout << "Number of files and catalogs: " << names.size() << endl;
    cout << "All " << endl;
    for(auto x:names)

```

```

        cout<<x.first<<" "<<x.second<<endl;
    cout<<"Input size: "<<endl; cin>>size0;
    cout << "Enter the number of threads you would like to start: ";
    int th; cin >> th;
    pthread_t id[th];
    per_thread = names.size()/th;
    for(int i=0;i<th;i++)
    {
        pthread_create(&(id[i]), NULL, sort, (void*)i);
    }
    sleep(0.1);

    cout << "Enter the number of process and the priority: ";
    cin >> num >> pr;

    for(int i=0;i<th;i++)
    {
        pthread_join(id[i],NULL);
    }
    cout<< "Input 0 to see results:"<<endl;
    int b; cin >> b;
    for(int i=0; i<res.size(); i++)
        cout<<res[i]<<endl;

    return 0;
}

```

## ***Results***

```
/home/ubuntu/CLionProjects/lab9/cmake-build-debug/untitled
Number of files and catalogs: 358
All
/home catalog
/home/ubuntu catalog
/home/ubuntu/.bash_history file
/home/ubuntu/.gnupg catalog
/home/ubuntu/.gnupg/pubring.kbx file
/home/ubuntu/.gnupg/private-keys-v1.d catalog
/home/ubuntu/.gnupg/trustdb.gpg file
/home/ubuntu/Шаблони catalog
/home/ubuntu/Зображення catalog
/home/ubuntu/Зображення/Знімок екрану з 2020-12-07 17-08-30.png file
/home/ubuntu/Зображення/Знімок екрану з 2020-12-07 11-21-27.png file
/home/ubuntu/Зображення/Знімок екрану з 2020-11-28 23-17-38.png file
/home/ubuntu/CLionProjects catalog
/home/ubuntu/CLionProjects/lab9 catalog
/home/ubuntu/CLionProjects/lab9/cmake-build-debug catalog
/home/ubuntu/CLionProjects/lab9/cmake-build-debug/CMakeCache.txt file
/home/ubuntu/CLionProjects/lab9/cmake-build-debug/untitled.cbp file
/home/ubuntu/CLionProjects/lab9/cmake-build-debug/cmake_install.cmake file
/home/ubuntu/CLionProjects/lab9/cmake-build-debug/Testing catalog
/home/ubuntu/CLionProjects/lab9/cmake-build-debug/untitled file
/home/ubuntu/CLionProjects/lab9/cmake-build-debug/Makefile file
/home/ubuntu/CLionProjects/lab9/cmake-build-debug/CMakeFiles catalog
/home/ubuntu/CLionProjects/lab9/CMakeLists.txt file
/home/ubuntu/CLionProjects/lab9/.idea catalog
/home/ubuntu/CLionProjects/lab9/.idea/.qitignore file
Problems  TODO  Terminal  Messages  CMake
```

І тд..)

```
Input size:
84
Enter the number of threads you would like to start: 4
Enter the number of process and the priority: Thread with ID 139699918194432 has started!
Interval limits from 268 356
Thread with ID 139699926587136 has started!
Interval limits from 179 267
Thread with ID 139699934979840 has started!
Interval limits from 90 178
Thread with ID 139699943372544 has started!
Interval limits from 1 89
1 10

Thread with ID 139935063222912 has ended!
Input 0 to see results:0
/home/ubuntu/.java/.userPrefs/jetbrains/syncsettings
/home/ubuntu/.config/ibus/bus

Process finished with exit code 0
```

Htop:

```
3546 ubuntu 20 0 39100 3272 2984 S 0.0 0.1 0:00.09 /home/ubuntu/CLionProjects/lab9/cmake-build-debug/un
3554 ubuntu 20 0 39100 3272 2984 S 0.0 0.1 0:00.00 /home/ubuntu/CLionProjects/lab9/cmake-build-debug
3553 ubuntu 20 0 39100 3272 2984 S 0.0 0.1 0:00.00 /home/ubuntu/CLionProjects/lab9/cmake-build-debug
3552 ubuntu 20 0 39100 3272 2984 S 0.0 0.1 0:00.00 /home/ubuntu/CLionProjects/lab9/cmake-build-debug
3551 ubuntu 35 15 39100 3272 2984 S 0.0 0.1 0:00.00 /home/ubuntu/CLionProjects/lab9/cmake-build-debug
help F2Setup F3Search F4Filter F5Sorted F6Collap F7Nice - F8Nice + F9Kill F10Quit
```

Висновок: у ході роботи над цією лабораторною, я реалізувала пошук файлів по розміру у декількох потоках, знайшла інтервали кожного з них та перевірила зміну пріоритету в утиліті htop.