

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
Факультет прикладної математики
Кафедра прикладної математики

Звіт
із лабораторної роботи №1
із дисципліни «Розподілені і хмарні обчислення»

Виконав:
студент групи КМ-01
Скорденко Д. О.

Керівник:
доцент кафедри ПМА
Ліскін В. О.

ЗМІСТ

Вступ	3
1 Основна частина	4
2 Опис програми [Тестовий приклад]	5
Висновки	7
Додаток Код лістинги	8

ВСТУП

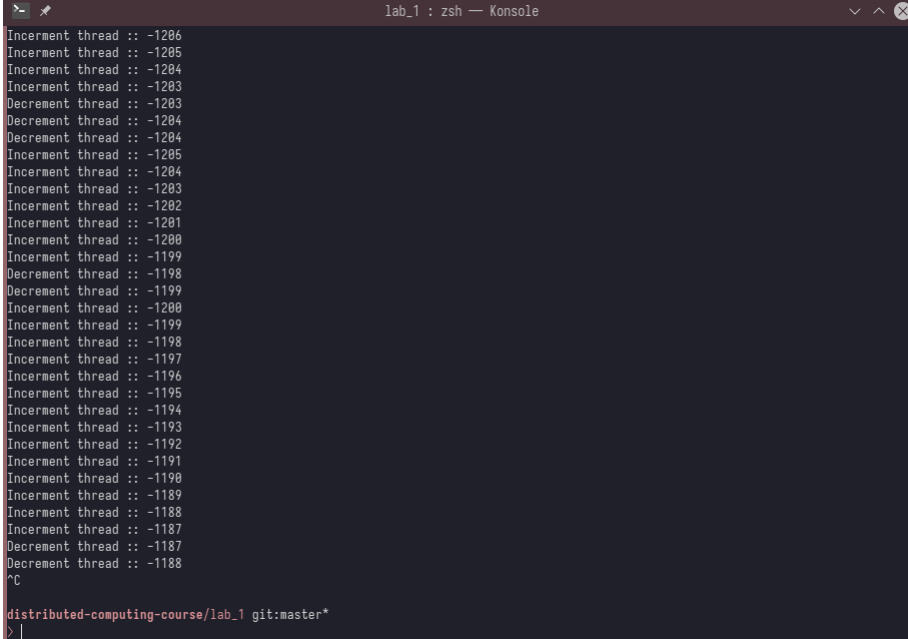
Мета: навчитись працювати із примітивом `thread` (потокom)

1 ОСНОВНА ЧАСТИНА

Опис програми: Для демонстрації роботи із потоками буде реалізовано програму інкремент/декримент. Кожна із операцій буде виконуватись в окремому потоці.

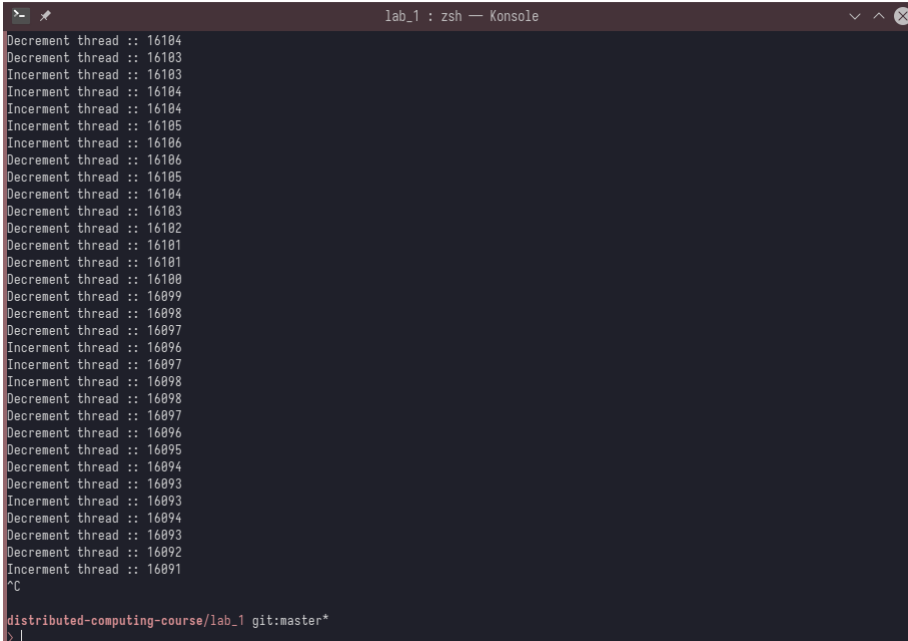
2 ОПИС ПРОГРАМИ [ТЕСТОВИЙ ПРИКЛАД]

Всі результати наводять роботу програми протягом 1-2 секунд.




```
lab_1 : zsh — Konsole
Increment thread :: -1206
Increment thread :: -1205
Increment thread :: -1204
Increment thread :: -1203
Decrement thread :: -1203
Decrement thread :: -1204
Decrement thread :: -1204
Increment thread :: -1205
Increment thread :: -1204
Increment thread :: -1203
Increment thread :: -1202
Increment thread :: -1201
Increment thread :: -1200
Increment thread :: -1199
Decrement thread :: -1198
Decrement thread :: -1199
Increment thread :: -1200
Increment thread :: -1199
Increment thread :: -1198
Increment thread :: -1197
Increment thread :: -1196
Increment thread :: -1195
Increment thread :: -1194
Increment thread :: -1193
Increment thread :: -1192
Increment thread :: -1191
Increment thread :: -1190
Increment thread :: -1189
Increment thread :: -1188
Decrement thread :: -1187
Decrement thread :: -1188
^C
distributed-computing-course/lab_1 git:master*
> |
```

Рисунок 2.1 – Відсутній пріоритет



```
lab_1 : zsh — Konsole
Decrement thread :: 16104
Decrement thread :: 16103
Increment thread :: 16103
Increment thread :: 16104
Increment thread :: 16104
Increment thread :: 16105
Increment thread :: 16106
Decrement thread :: 16106
Decrement thread :: 16105
Decrement thread :: 16104
Decrement thread :: 16103
Decrement thread :: 16102
Decrement thread :: 16101
Decrement thread :: 16101
Decrement thread :: 16100
Decrement thread :: 16099
Decrement thread :: 16098
Decrement thread :: 16097
Increment thread :: 16096
Increment thread :: 16097
Increment thread :: 16098
Decrement thread :: 16098
Decrement thread :: 16097
Decrement thread :: 16096
Decrement thread :: 16095
Decrement thread :: 16094
Decrement thread :: 16093
Increment thread :: 16093
Decrement thread :: 16094
Decrement thread :: 16093
Decrement thread :: 16092
Increment thread :: 16091
^C
distributed-computing-course/lab_1 git:master*
> |
```

Рисунок 2.2 – Мінімальний пріоритет інкременту



```
lab_1 : zsh — Konsole
Increment thread :: -19144
Increment thread :: -19144
Increment thread :: -19143
Decrement thread :: -19142
Decrement thread :: -19143
Increment thread :: -19143
Increment thread :: -19142
Decrement thread :: -19142
Decrement thread :: -19143
Increment thread :: -19144
Increment thread :: -19143
Decrement thread :: -19143
Decrement thread :: -19144
Decrement thread :: -19144
Increment thread :: -19145
Increment thread :: -19144
Decrement thread :: -19144
Decrement thread :: -19145
Decrement thread :: -19145
Increment thread :: -19146
Increment thread :: -19145
Decrement thread :: -19144
Decrement thread :: -19145
Increment thread :: -19145
Increment thread :: -19144
Increment thread :: -19144
Decrement thread :: -19143
Decrement thread :: -19144
Increment thread :: -19144
Increment thread :: -19143
Decrement thread :: -19142
^C
distributed-computing-course/lab_1 git:master* 6s
> |
```

Рисунок 2.3 – Мінімальний пріорітет декременту

ВИСНОВКИ

Було реалізовано програму інкремент/декремент для демонстрації роботи потоків та їх пріоритезації.

Додаток

Код лістинги

*Примітка: У код лістингах при копіюванні втрачається форматування (не копіюються пробіли). Файли прикріплено до цього pdf (вкладка "прикріплені файли").

Listing 1: main.rs

```
use std::{thread};
use std::sync::{Arc, Mutex};
use thread_priority::*;

fn increment(refr: &Arc<Mutex<i32>>) {
    let arc = Arc::clone(&refr);
    let mut num = arc.lock().unwrap();
    *num += 1;
}

fn decrement(refr: &Arc<Mutex<i32>>) {
    let arc = Arc::clone(&refr);
    let mut num = arc.lock().unwrap();
    *num -= 1;
}

fn main() {
```



```

let shared: Arc<Mutex<i32>> = Arc::new(Mutex::new(0));

let increment_shared = shared.clone();
let handle_increment = thread::spawn(move || {
    //assert!(set_current_thread_priority(ThreadPriority::Min).is_ok())
    loop {
        increment(&increment_shared);
        let num = increment_shared.lock().unwrap();
        println!("Increment thread :: {num}");
    }
});

let decrement_shared = shared.clone();
let handle_decrement = thread::spawn(move || {
    //assert!(set_current_thread_priority(ThreadPriority::Min).is_ok())
    loop {
        decrement(&decrement_shared);
        let num = decrement_shared.lock().unwrap();
        println!("Decrement thread :: {num}");
    }
});

handle_increment.join().unwrap();
handle_decrement.join().unwrap();
}

```