

Proyecto 1

Paralelismo en un ambiente
controlado

- Miguel Vásquez Bojanini
- Manuel Villegas Michel
- Esteban Muriel Roldan

Objetivo

Familiarizarse con la programación en paralelo utilizando diferentes procesos y como podemos manipularlos e ubicarlos para afectar su eficiencia dentro de la maquina. Además de manejar con grandes volúmenes de información.

Características de los dispositivos

Razer Balade 2018

- **Arquitectura:** x86_64
- **CPU:** Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz
 - 12 cores (0-11)
 - Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz, 900 MHz
 - 2 hilos por core
- **OS:** EndeavourOS (Linux)
- **RAM:** 16G



ASUS zenbook

- **Arquitectura:** x86_64
- **CPU:** Intel m3-6Y30 (4) @ 2.200GHz
 - 4 cores (0-3)
 - Hilos por core: 2
- **OS:** Arch (Linux)
- **RAM:** 8G



Steam Deck

- **Arquitectura:** x86_64
- **CPU:** AMD Zen 2 semi-custom
 - 8 cores
 - 8 hilos
 - 2.4 – 3.5 GHz
- **OS:** SteamOS (Arch Linux)
- **RAM:** 16GB



Ejecución



Ejecucion continua

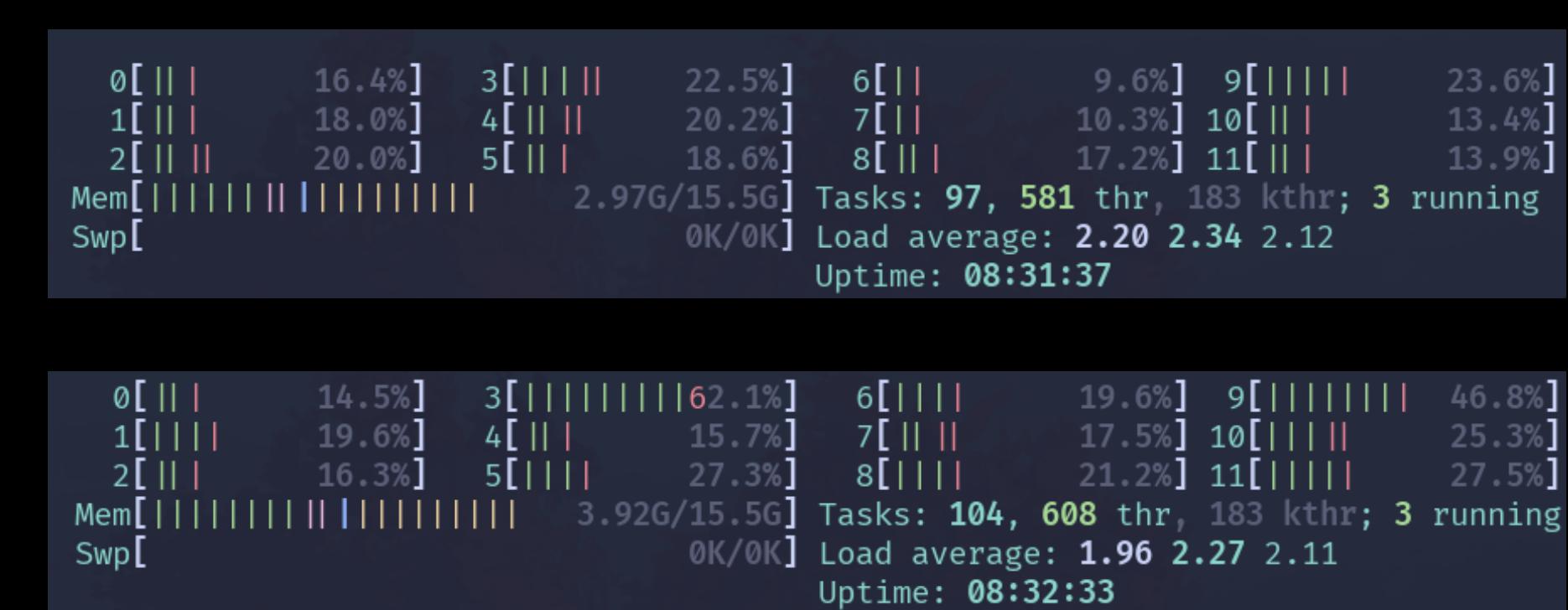
Razer Balade 2018

```
.... Multirprocessing/src [!] main[x!?][☕ v21.0.3]
:: java App -f ./output/
Start time: 16:25:50.515685221

Reading file: KRvideos.csv
Time in process (millis): 82
Reading file: MXvideos.csv
Time in process (millis): 19
Reading file: USvideos.csv
Time in process (millis): 982
Reading file: GBvideos.csv
Time in process (millis): 583
Reading file: INvideos.csv
Time in process (millis): 646
Reading file: FRvideos.csv
Time in process (millis): 523
Reading file: JPvideos.csv
Time in process (millis): 26
Reading file: DEvideos.csv
Time in process (millis): 670
Reading file: RUVideos.csv
Time in process (millis): 29
Reading file: CAvideos.csv
Time in process (millis): 719
Execute total time: 4287 milliseconds

Final time: 16:25:54.812546089
```

Tiempo de lectura en cada archivo



Pre Ejecucion

Tiempo de la ejecución total

Durante ejecución

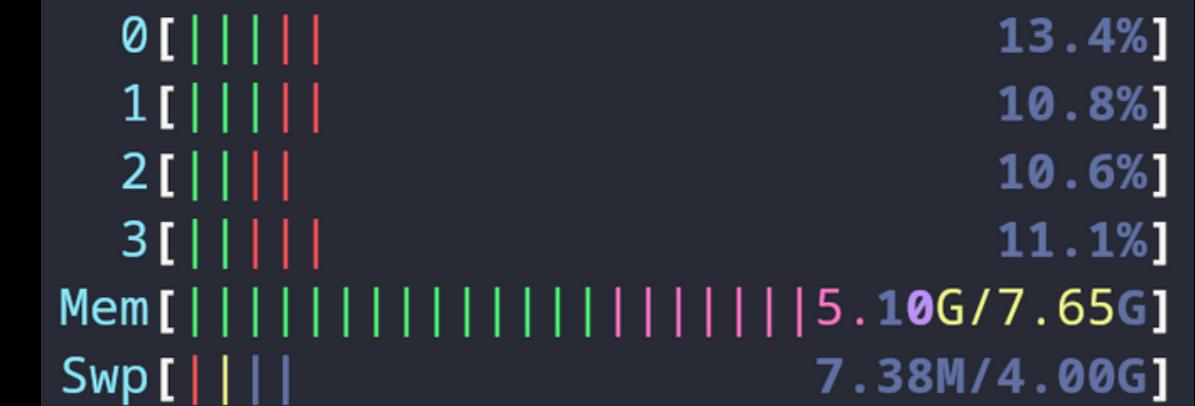
Ejecucion continua

ASUS zenbook

```
Start time: 20:31:51.015591599
Reading file: USvideos.csv
Time in process (millis): 2094
Reading file: INvideos.csv
Time in process (millis): 1574
Reading file: DEvideos.csv
Time in process (millis): 2066
Reading file: MXvideos.csv
Time in process (millis): 13
Reading file: RUvideos.csv
Time in process (millis): 134
Reading file: JPvideos.csv
Time in process (millis): 106
Reading file: CAvideos.csv
Time in process (millis): 1705
Reading file: KRvideos.csv
Time in process (millis): 19
Reading file: FRvideos.csv
Time in process (millis): 1454
Reading file: GBvideos.csv
Time in process (millis): 1240
Execute total time: 10415 milliseconds
Final time: 20:32:01.829085131
```

Tiempo de lectura en cada archivo

Pre Ejecucion



Durante ejecución

Tiempo de la ejecución total

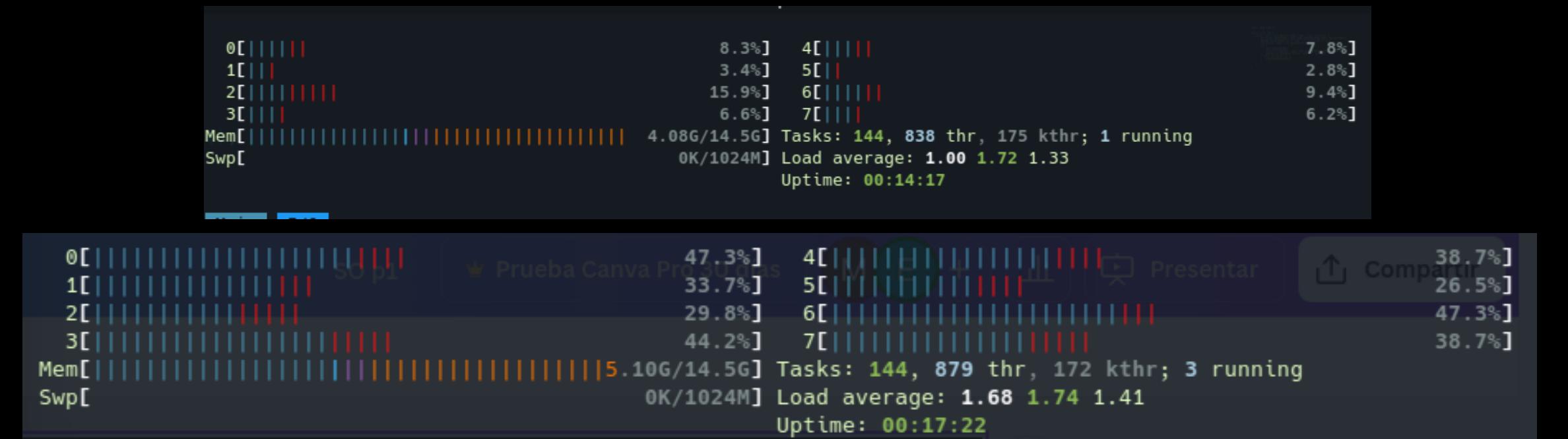
Ejecución continua

Steam Deck

```
● bash-5.2$ java App -f ./output/
Start time: 20:01:59.477300572

Reading file: FRvideos.csv
Time in process (millis): 1242
Reading file: JPvideos.csv
Time in process (millis): 51
Reading file: CAvideos.csv
Time in process (millis): 865
Reading file: USvideos.csv
Time in process (millis): 839
Reading file: KRvideos.csv
Time in process (millis): 7
Reading file: GBvideos.csv
Time in process (millis): 626
Reading file: MXvideos.csv
Time in process (millis): 5
Reading file: DEvideos.csv
Time in process (millis): 772
Reading file: INvideos.csv
Time in process (millis): 692
Reading file: RUvideos.csv
Time in process (millis): 33
Execute total time: 5143 milliseconds
Final time: 20:02:04.647616463
```

Tiempo de lectura en cada archivo



Antes de ejecución

En ejecución

Tiempo de la ejecución total

Solucion

```
static final int PAGE_LIMIT = 4096; // Each char equals 2 bytes; 4k = 4000 bytes

public static void main(String[] args) { 0 references
    Instant start = Instant.now();
    List<List<String>> list = new ArrayList<>();
    int limit = PAGE_LIMIT;
    List<String> currentData = new ArrayList<>();
    try {
        File myObj = new File(args[0]);
        // File myObj = new File("./src/output/MXvideos.csv");
        Scanner myReader = new Scanner(myObj);
        while (myReader.hasNextLine()) {
            String line = myReader.nextLine();
            while (true) {
                if (line.getBytes(StandardCharsets.UTF_8).length == limit) {
                    currentData.add(line);
                    list.add(currentData);
                    currentData = new ArrayList<>();
                    limit = PAGE_LIMIT;
                    break;
                } else if (line.getBytes().length < limit) {
                    currentData.add(line);
                    limit -= line.getBytes().length;
                    break;
                } else {
                    byte[] sub = Arrays.copyOf(line.getBytes(), limit);
                    currentData.add(new String(sub, StandardCharsets.UTF_8));
                    list.add(currentData);
                    currentData = new ArrayList<>();
                    line = line.substring(new String(sub, StandardCharsets.UTF_8).length());
                    limit = PAGE_LIMIT;
                }
            }
        }
        if (currentData.get(0).length() > 0) {
            list.add(currentData);
        }
        myReader.close();
    }
```

Lectura y almacenamiento en
“paginas” de 4k

Flag -s

Razer Balade 2018

```
••• Multirprocessing/src [!] main[x!?][⌚ v21.0.3][⌚ 24s]
:: java App -s -f ./output/
Start time: 16:38:01.879720957

First file load time: 16:38:01.906273371 → Carga del primer archivo
Process with PID: 4005444 File: ./output/KRvideos.csv Finish time: 16:38:03.257562696
  Time in process (millis): 271
Process with PID: 4005935 File: ./output/MXvideos.csv Finish time: 16:38:04.680547795
  Time in process (millis): 215
Process with PID: 4006371 File: ./output/USvideos.csv Finish time: 16:38:07.714321263
  Time in process (millis): 1808
Process with PID: 4007320 File: ./output/GBvideos.csv Finish time: 16:38:10.400872515
  Time in process (millis): 1548
Process with PID: 4008220 File: ./output/INvideos.csv Finish time: 16:38:12.890547850
  Time in process (millis): 1435
Process with PID: 4009039 File: ./output/FRvideos.csv Finish time: 16:38:15.849321438
  Time in process (millis): 1848
Process with PID: 4010023 File: ./output/JPvideos.csv Finish time: 16:38:17.361006612
  Time in process (millis): 424
Process with PID: 4010520 File: ./output/DEvideos.csv Finish time: 16:38:20.123338183
  Time in process (millis): 1715
Process with PID: 4011455 File: ./output/RUvideos.csv Finish time: 16:38:21.817568743
  Time in process (millis): 609
Last file load time: 16:38:21.850998724 → Carga del ultimo archivo
Process with PID: 4012004 File: ./output/CAvideos.csv Finish time: 16:38:24.649800199
  Time in process (millis): 1789
Execute total time: 22785 milliseconds
Final time: 16:38:24.678638525
```

Pre Ejecucion

```
0[|||] 14.2% 3[||||] 28.8% 6[|||] 18.6% 9[|||] 7.7%
1[|||] 18.5% 4[|||] 18.2% 7[|||] 21.9% 10[|||] 10.1%
2[|||] 17.2% 5[|||] 13.8% 8[|||] 5.9% 11[|||] 14.2%
Mem[|||||] 2.90G/15.5G Tasks: 91, 477 thr, 189 kthr; 2 running
Swp[0K/0K] Load average: 2.62 2.32 2.15
Uptime: 08:43:09
```

Durante ejecución

```
0[|||] 17.3% 3[|||] 12.7% 6[|||] 19.6% 9[||||] 26.2%
1[|||] 14.6% 4[||||] 32.5% 7[|||] 18.2% 10[|||] 20.8%
2[|||] 10.4% 5[|||] 14.3% 8[|||||] 99.4% 11[|||] 10.3%
Mem[|||||] 3.00G/15.5G Tasks: 93, 508 thr, 189 kthr; 6 running
Swp[0K/0K] Load average: 2.73 2.37 2.18
Uptime: 08:44:00
```

Core en uso

Carga del ultimo archivo

Flag -s

ASUS zenbook

```
Start time: 20:43:16.710822096
First file load time: 20:43:17.151149956
Process with PID: 4402File: ./output/USvideos.csv Finish time: 20:43:22.1886369
85 Time in process (millis): 3257
Process with PID: 4419File: ./output/INvideos.csv Finish time: 20:43:28.5297931
28 Time in process (millis): 4511
Process with PID: 4455File: ./output/DEvideos.csv Finish time: 20:43:35.8782493
91 Time in process (millis): 5397
Process with PID: 4476File: ./output/MXvideos.csv Finish time: 20:43:40.0637678
16 Time in process (millis): 605
Process with PID: 4491File: ./output/RUvideos.csv Finish time: 20:43:43.9158891
95 Time in process (millis): 1231
Process with PID: 4508File: ./output/JPvideos.csv Finish time: 20:43:48.0052813
12 Time in process (millis): 1448
Process with PID: 4523File: ./output/CAvideos.csv Finish time: 20:43:55.7612075
78 Time in process (millis): 5263
Process with PID: 4560File: ./output/KRvideos.csv Finish time: 20:43:59.1412284
62 Time in process (millis): 403
Process with PID: 4576File: ./output/FRvideos.csv Finish time: 20:44:04.8322421
06 Time in process (millis): 3354
Last file load time: 20:44:04.883129206
Process with PID: 4597File: ./output/GBvideos.csv Finish time: 20:44:11.3735064
98 Time in process (millis): 4729
Execute total time: 54303 milliseconds
Final time: 20:44:11.425846290
```

Carga del primer archivo

Pre Ejecucion



Core en uso

Durante ejecución

Carga del ultimo archivo

Flag -s

Steam Deck

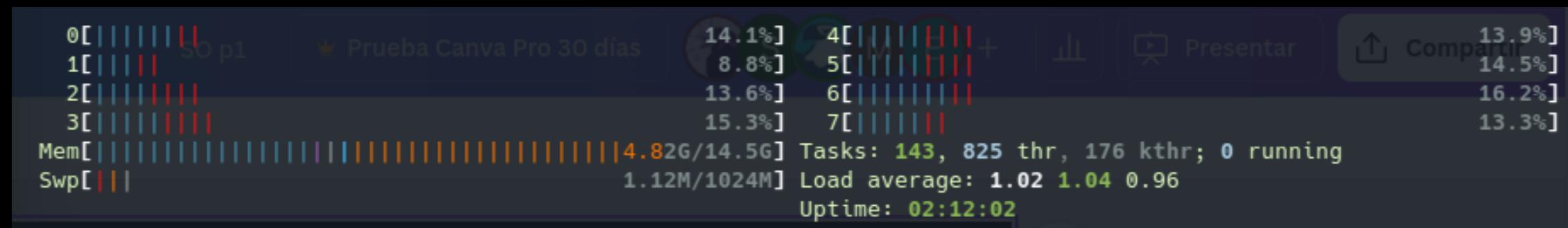
Carga del primer archivo

```
● bash-5.2$ java App -s -f ./output/
Start time: 20:04:15.580830015

First file load time: 20:04:15.645796442
Process with PID: 926File: ./output/FRvideos.csv Finish time: 20:04:19.102150353 Time in process (millis): 2001
Process with PID: 958File: ./output/JPvideos.csv Finish time: 20:04:20.902587828 Time in process (millis): 500
Process with PID: 974File: ./output/CAvideos.csv Finish time: 20:04:24.504366366 Time in process (millis): 2268
Process with PID: 1005File: ./output/USvideos.csv Finish time: 20:04:27.853569418 Time in process (millis): 2073
Process with PID: 1029File: ./output/KRvideos.csv Finish time: 20:04:29.391143776 Time in process (millis): 251
Process with PID: 1051File: ./output/GBvideos.csv Finish time: 20:04:32.988661514 Time in process (millis): 2228
Process with PID: 1075File: ./output/MXvideos.csv Finish time: 20:04:34.646996567 Time in process (millis): 345
Process with PID: 1097File: ./output/DEvideos.csv Finish time: 20:04:37.789503858 Time in process (millis): 1749
Process with PID: 1121File: ./output/INvideos.csv Finish time: 20:04:40.918029920 Time in process (millis): 1626
Last file load time: 20:04:40.984281492
Process with PID: 1143File: ./output/RUvideos.csv Finish time: 20:04:42.977580948 Time in process (millis): 659
Execute total time: 27415 milliseconds

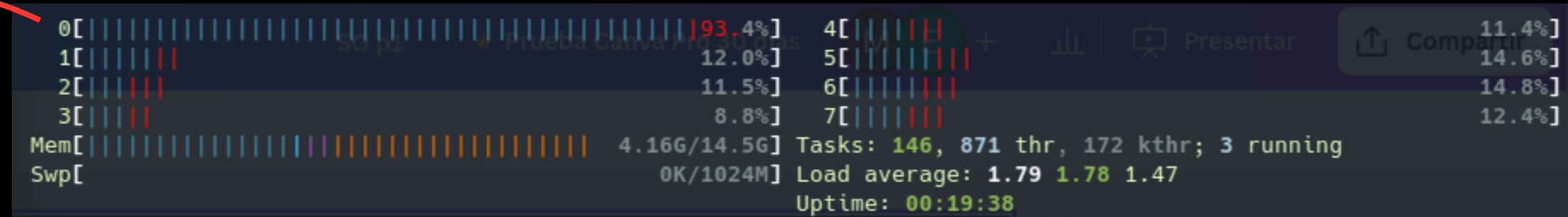
Final time: 20:04:43.029430749
```

Carga del ultimo archivo



Pre Ejecucion

Core en uso



Durante ejecución

Solucion

Con el comando “ps” obtenemos el core actual en el que se está corriendo el proceso inicial/padre

```
private String getCore(long pid) throws IOException {
    // Get parent PID and current core
    String[] commands = { "ps", "-o", "psr", "-p", "" + pid };
    Process proc = Runtime.getRuntime().exec(commands);
    BufferedReader stdInput = new BufferedReader(new InputStreamReader(proc.getInputStream()));

    // Read the output from the command
    String s;
    String core = "0";
    while ((s = stdInput.readLine()) != null) {
        if (!s.equals("PSR".strip())) {
            core = s.strip();
        }
    }
    return core;
}
```

```
private int pathS() throws IOException {
    long mainId = ProcessHandle.current().pid();
    for (int i = 0; i < this.pathsLength; i++) {
        String core = getCore(mainId);
        // Create and start new process
        ProcessBuilder pb = new ProcessBuilder(...command:"taskset", "-c", core, "java", "./Reader.java",
            manager.getFolder() + manager.getPaths()[i]);
        Process p = pb.start();
        if (i == 0) {
            LocalTime firtstFile = java.time.LocalDateTime.now().toLocalTime();
            System.out.println("First file load time: " + firtstFile);
        }
        if (i == this.pathsLength - 1) {
            LocalTime lastFile = java.time.LocalDateTime.now().toLocalTime();
            System.out.println("Last file load time: " + lastFile);
        }
        printData(p);
        p.destroy();
    }
    return 0;
}
```

Con Taskset se le indica al proceso en cuál core correr

Reader.java: clase encargada de leer los archivos

Flag -m

Razer Balade 2018

```
••• Multirprocessing/src [!] main[x!?][☕ v21.0.3][⌚ 26s]
:: java App -m -f ./output/
Start time: 16:42:44.090735455

First file load time: 16:42:44.113599348 → Carga del primer archivo
Process with PID: 4103383 File: ./output/KRvideos.csv Finish time: 16:42:44.762055022
  Time in process (millis): 84
Process with PID: 4103598 File: ./output/MXvideos.csv Finish time: 16:42:45.462585230
  Time in process (millis): 79
Process with PID: 4103805 File: ./output/USvideos.csv Finish time: 16:42:47.243825894
  Time in process (millis): 1174
Process with PID: 4104359 File: ./output/GBvideos.csv Finish time: 16:42:48.819876279
  Time in process (millis): 933
Process with PID: 4104817 File: ./output/INvideos.csv Finish time: 16:42:50.393329294
  Time in process (millis): 952
Process with PID: 4105271 File: ./output/FRvideos.csv Finish time: 16:42:51.950602028
  Time in process (millis): 972
Process with PID: 4105744 File: ./output/JPvideos.csv Finish time: 16:42:52.760878935
  Time in process (millis): 192
Process with PID: 4105971 File: ./output/DEvideos.csv Finish time: 16:42:54.502796563
  Time in process (millis): 1089
Process with PID: 4106485 File: ./output/RUvideos.csv Finish time: 16:42:55.376586080
  Time in process (millis): 228
Last file load time: 16:42:55.389640907 → Carga del ultimo archivo
Process with PID: 4106755 File: ./output/CAvideos.csv Finish time: 16:42:57.120780360
  Time in process (millis): 1140
Execute total time: 13037 milliseconds

Final time: 16:42:57.139209777
```

Pre Ejecucion

```
0[||| 11.1%] 3[||| 14.3%] 6[||| 14.7%] 9[||| 7.1%
1[||| 11.8%] 4[||| 13.6%] 7[||| 11.0%] 10[||| 12.0%
2[||| 19.9%] 5[||| 12.1%] 8[||| 12.4%] 11[||| 5.8%
Mem[||||| 2.96G/15.5G] Tasks: 91, 477 thr, 174 kthr; 2 running
Swp[ 0K/0K] Load average: 2.32 2.41 2.23
Uptime: 08:47:18
```

```
0[||| 22.1%] 3[||||| 39.7%] 6[||||| 87.2%] 9[||||| 47.7%
1[||||| 39.9%] 4[||||| 40.8%] 7[||||| 55.8%] 10[|||| 23.9%
2[||||| 46.5%] 5[|||| 38.6%] 8[||||| 44.9%] 11[|||| 28.2%
Mem[||||| 3.02G/15.5G] Tasks: 93, 520 thr, 175 kthr; 8 running
Swp[ 0K/0K] Load average: 2.20 2.37 2.22
Uptime: 08:47:54
```

Durante ejecución

Carga del ultimo archivo

Flag -m

ASUS zenbook

```
Start time: 20:47:31.364872129

First file load time: 20:47:31.839985911
^[[2;10~Process with PID: 4882File: ./output/USvideos.csv Finish time: 20:47:35
.463918662 Time in process (millis): 2303
Process with PID: 4910File: ./output/INvideos.csv Finish time: 20:47:38.9210309
08 Time in process (millis): 2308
Process with PID: 4935File: ./output/DEvideos.csv Finish time: 20:47:42.8725292
81 Time in process (millis): 2877
Process with PID: 4978File: ./output/MXvideos.csv Finish time: 20:47:44.1930040
36 Time in process (millis): 186
Process with PID: 5003File: ./output/RUvideos.csv Finish time: 20:47:45.8507669
18 Time in process (millis): 569
Process with PID: 5025File: ./output/JPvideos.csv Finish time: 20:47:47.8162837
35 Time in process (millis): 662
Process with PID: 5073File: ./output/CAvideos.csv Finish time: 20:47:52.3547530
46 Time in process (millis): 3160
Process with PID: 5099File: ./output/KRvideos.csv Finish time: 20:47:53.8916677
97 Time in process (millis): 271
Process with PID: 5122File: ./output/FRvideos.csv Finish time: 20:47:57.7104523
52 Time in process (millis): 2273
Last file load time: 20:47:57.732664793
Process with PID: 5155File: ./output/GBvideos.csv Finish time: 20:48:01.6072982
68 Time in process (millis): 2620
Execute total time: 29851 milliseconds

Final time: 20:48:01.679195195
```

Carga del primer archivo

Pre Ejecucion



Carga del ultimo archivo

Durante ejecución

Flag -m

Steam deck

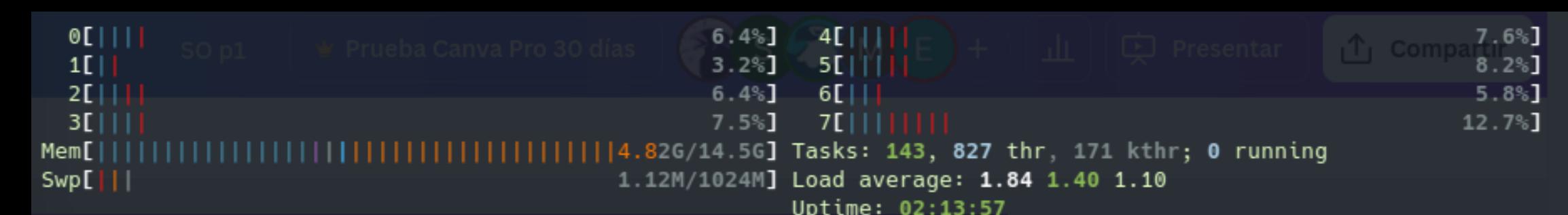
```
● bash-5.2$ java App -m -f ./output/
Start time: 20:07:40.565218431

First file load time: 20:07:40.619040980
Process with PID: 1235File: ./output/FRvideos.csv Finish time: 20:07:42.633091153 Time in process (millis): 1095
Process with PID: 1277File: ./output/JPvideos.csv Finish time: 20:07:43.487966116 Time in process (millis): 219
Process with PID: 1303File: ./output/CAvideos.csv Finish time: 20:07:45.477964827 Time in process (millis): 1302
Process with PID: 1345File: ./output/USvideos.csv Finish time: 20:07:47.471741295 Time in process (millis): 1296
Process with PID: 1373File: ./output/KRvideos.csv Finish time: 20:07:48.239282289 Time in process (millis): 107
Process with PID: 1398File: ./output/GBvideos.csv Finish time: 20:07:49.926248550 Time in process (millis): 1044
Process with PID: 1433File: ./output/MXvideos.csv Finish time: 20:07:50.681173428 Time in process (millis): 89
Process with PID: 1464File: ./output/DEvideos.csv Finish time: 20:07:52.613124013 Time in process (millis): 1235
Process with PID: 1492File: ./output/INVideos.csv Finish time: 20:07:54.529338086 Time in process (millis): 1173
Last file load time: 20:07:54.552553763
Process with PID: 1529File: ./output/RUvideos.csv Finish time: 20:07:55.524195437 Time in process (millis): 268
Execute total time: 14956 milliseconds

Final time: 20:07:55.551241865
```

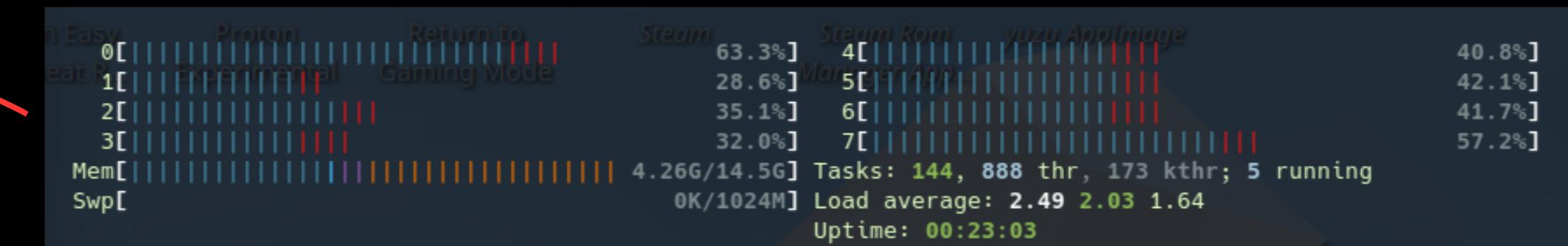
Carga del primer archivo

Pre Ejecucion



Carga del ultimo archivo

Cores en uso



Durante ejecución

Solucion

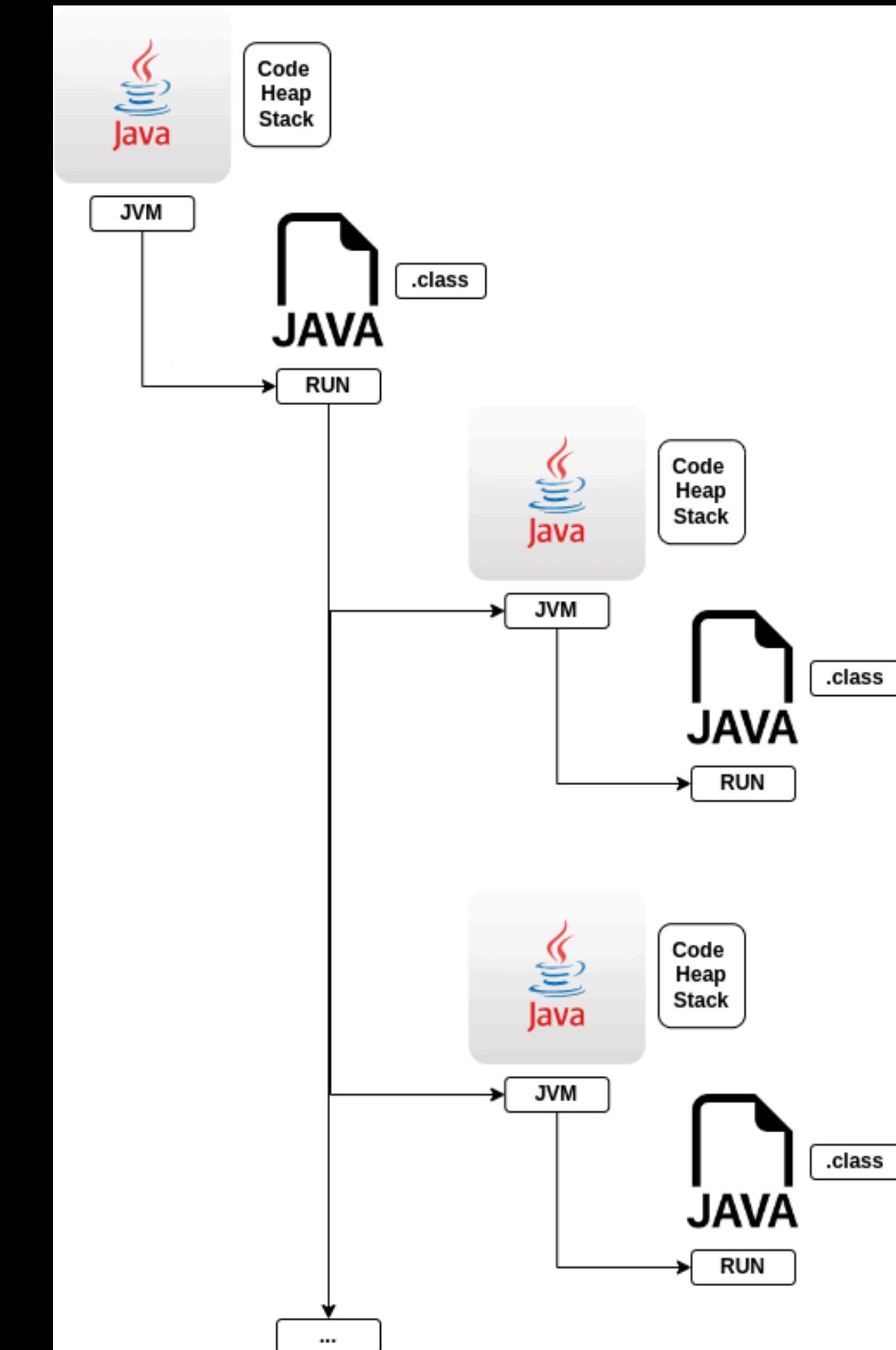
```
private int pathM() throws IOException {
    for (int i = 0; i < this.pathsLength; i++) {
        ProcessBuilder pb = new ProcessBuilder(...command:"java", "./Reader.java",
        |   |   manager.getFolder() + manager.getPaths()[i]);
        Process p = pb.start();

        if (i == 0) {
            LocalTime firtstFile = java.time.LocalDateTime.now().toLocalTime();
            System.out.println("First file load time: " + firtstFile);
        }
        if (i == this.pathsLength - 1) {
            LocalTime lastFile = java.time.LocalDateTime.now().toLocalTime();
            System.out.println("Last file load time: " + lastFile);
        }
        printData(p);
        p.destroy();
    }
    return 0;
}
```

Sin el comando Taskset, los procesos son libres de correr en los Cores disponibles al momento de crearse

Consideraciones

- Las operaciones de IO son lo que mas puede demorar la ejecución de los programas, ya que están interrumpiendo la ejecución y dándole control al sistema operativo.
- Java, al ser un lenguaje que necesita de una maquina virtual para ser ejecutado necesita alojar una maquina virtual en memoria por cada proceso. Lo cual aumenta el uso de memoria y demora su ejecución. **Mas en la siguiente diapositiva.**
- Los comandos son **independientes del sistema**, como los que utilizamos a la hora de especificar la afinidad de un proceso a un Core en específico solo están disponibles en distribuciones de Linux. (Taskset y ps).



Conclusiones

- En la ejecución secuencial evidenciamos que el uso de la memoria RAM es mayor, porque toda la información esta siendo almacenada en una unica estructura de datos.
- La virtualización de los procesos tiene la ventaja de poder ejecutar varios procesos en distintas CPUs pero son costosos en cuanto a recursos y tiempo .
- Los cambios de contexto se hacen bastante costosos y la comunicación entre los procesos es mas compleja porque no comparten memoria (IPC).
- Application Isolation API (JSR 121) nunca fue implementada.

Webgrafia

- <https://www.geeksforgeeks.org/computer-organization-amdahls-law-and-its-proof/>
- https://en.wikipedia.org/wiki/Amdahl%27s_law
- https://www.golinuxcloud.com/java-multiprocessing/#Introduction_to_Java_Multiprocessing

Gracias

