

Taller de Sistemas Operativos

Reporte técnico taller 01

Escuela de Ingeniería Informática

Universidad de Valparaíso

Sebastián González Morales

Sebastian.gonzalez@alumnos.uv.cl

1 Introducción

Este artículo principalmente explicará el diseño, la implementación y resultados de un script que permitirá realizar estadísticas descriptivas de datos sobre cierta simulación. En esta ocasión se utilizará Bash y para el ingreso de parámetro se utilizará el estándar Unix.

GNU Bash o simplemente Bash (Bourne-again shell) es un lenguaje de comandos y shell de Unix escrito por Brian Fox para el Proyecto GNU como un reemplazo de software libre para el shell Bourne. Bash es un procesador de comandos que generalmente se ejecuta en una ventana de texto donde el usuario escribe comandos que causan acciones. Bash también puede leer y ejecutar comandos desde un archivo, llamado script de shell. [1]

Single UNIX Specification es el estándar en el que se miden las interfaces principales de un sistema operativo UNIX. El estándar UNIX incluye un amplio conjunto de características, y sus volúmenes principales son simultáneamente el estándar de interfaz de sistema operativo portátil IEEE (POSIX) y el estándar ISO / IEC 9945. La especificación abarca el entorno del sistema operativo base, los servicios de red, los servicios del sistema de ventanas y los aspectos de internacionalización y lenguajes de programación. [2]

Este documento se estructura de la siguiente forma, en la sección 2 se presentará una descripción del problema, ahí se describirán como están constituidos los archivos de texto y los datos a procesar. Luego en la sección 3 se explicará el diseño de la solución. En la sección 4 se presentará la metodología para llevar a cabo la implementación. En la sección 5, se presentará pruebas de ejecución del script desde otros directorios. Luego en la sección 6 se mostrarán los resultados obtenidos. En la sección 7 se dará una conclusión sobre el trabajo realizado y finalmente en la sección 8, están las referencias que se utilizaron.

2 Descripción del problema

Diseñar un script en Bash con el nombre **stats.sh**, el cual debe cumplir las siguientes tareas:

- 1) Determinar la cantidad máxima, mínima y el promedio para las siguientes métricas de desempeño computacional del simulador:
 - Tiempo de simulación total.

- Este se define de la siguiente forma: $\text{time execMakeAgents} + \text{timeExecCal} + \text{timeExecSim}$.
Memoria utilizada por el simulador.

Los resultados se entregarán en un archivo **metric.txt** con la estructura que se muestra en la tabla 1.

Tabla 1- Estructura para estadísticas de desempeño simulador.

```
tsimTotal:promedio:min:max
memUsed:promedio:min:max
```

2) Determinar el tiempo promedio de evacuación, además del mínimo y el máximo para los siguientes grupos de personas:

- Todas las personas simuladas.
- Sólo los residentes.
- Sólo visitantes tipo I.
- Sólo residentes, separados por grupo etario.
- Sólo Visitante Tipo I, separados por grupo etario.

Los resultados se deben entregar en un archivo llamado **evacuation.txt**. Utilizando la estructura del archivo que se muestra en la tabla 2.

Tabla 2.-Estructura para estadísticas personas.

```
alls:promedio:min:max
residents:promedio:min:max
visitorsI: promedio:min:max
residents-G0:promedio:min:max
residents-G1:promedio:min:max
residents-G2:promedio:min:max
residents-G3:promedio:min:max
visitorsI-G0:promedio:min:max
visitorsI-G1:promedio:min:max
visitorsI-G2:promedio:min:max
visitorsI-G3:promedio:min:max
```

3) Además de lo anterior se debe determinar el promedio de uso de teléfonos móviles, además del mínimo y el máximo para los instantes de tiempo especificados en los archivos usePhone-NNN.txt. Los resultados se deben entregar en un archivo **usePhone-stats.txt**, con la estructura de la tabla 3 por cada línea.

Tabla 3.- Estructura para estadísticas de uso de teléfono móvil.

timestamp:promedio:min:max

2.1 Contexto de los datos a utilizar

Los datos a procesar corresponden a estadísticas que provienen de cierta cantidad de experimentos de simulación de un sistema de evacuación de personas. Cada experimento es una simulación de una evacuación costera de la ciudad de Iquique ante un eventual tsunami, que considera 75000 personas. Existen tres modelos de personas: Residentes y Visitantes Tipo I y Visitantes Tipo II (Tabla 4). Todos los residentes conocen su zona de seguridad. Los visitantes, al inicio del proceso de evacuación, no. Eventualmente, un visitante puede pedir ayuda y conocer su zona de seguridad (Visitante tipo I). Si un visitante no logra determinar su zona de seguridad, queda caminando por la ciudad y no llega nunca a una zona de seguridad (Visitante tipo II).

Tabla 4.- Tipo de Personas simuladas

Tipo de Persona	Descripción
Residente	Persona que vive en la ciudad y conoce su zona segura
Visitante Tipo I	Visitante que durante la evacuación logra determinar su zona segura.
Visitante Tipo II	Visitante que durante la evacuación logra no determinar su zona segura.

Además de clasificación mencionada, cada persona pertenece a un grupo etario, los que se resumen en la Tabla 5.

Tabla 5.- Grupos etarios utilizados

Grupo Etario	Intervalo de edad
G0	0-14
G1	15-29
G2	30-64
G3	65 o más

2.2 Organización de los datos.

Cada simulación entrega los resultados ordenados en una estructura de directorios, según el esquema que se visualiza en la Figura 1, donde NNN es el identificador de la simulación, representado por un código numérico de tres dígitos.



Figura 1.- Estructura de directorios de los resultados de cada simulación.

2.3 Descripción de los datos

2.3.1 Archivo executionSummary.txt

```
numExperiment:tsim:calibrationTime:Residents:Visitors:timeExecMakeAgents:timeExecCal:timeExecSim:maxMemory:agentsMem  
0:3600:100:69000:6000:28252:33157:182800:288236:33648
```

Figura 2.- Estructura archivo executionSummary.txt

El archivo executionSummary-NNN.txt contiene datos sobre el desempeño de la simulación NNN. Posee una cabecera y una segunda línea con los datos. Ésta tiene 10 campos, separados por el símbolo ‘:’. En la Figura 2 se muestra el contenido del archivo que corresponde a la simulación 000. En la Tabla 6 se detallan los campos y su respectiva descripción.

Tabla 6.- Descripción de los campos del archivo executionSummary

Campo	Descripción	Ejemplo
numExperiment	Número del experimento.	0 (Corresponde al código 000)
Tsim	Tiempo de simulación.	3600: segundos de tiempo real que se simuló.
CalibrationTime	Tiempo de calibración del simulador.	100: segundo de tiempo de real que se destina a la calibración inicial de las personas.
Residents	Cantidad de Residentes simulados.	69000
Visitors	Cantidad de Visitantes simulados.	6000

timeExecMakeAgents	Tiempo real que demora en crear en memoria las personas simuladas.	28252 : milisegundos
timeExecCal	Tiempo real que demora la calibración de las personas.	33157 : milisegundos
timeExecSim	Tiempo real que toma la simulación.	182800 : milisegundo
maxMemory	Costo espacial del simulador.	288236 : KbBtes
agentsMem	Memoria utilizada por las estructuras de datos relacionadas con las personas.	33648 : kBites

2.3.2 Archivo summary.txt

```

numExperiment:id:model:groupAge:safeZone:distanceToTargetPos:responseTime:evacTime
0:0:0:2:Z1:8.271001:262.785961:1450.000000
0:1:0:1:Z2:7.345218:287.944671:1280.000000
0:2:0:0:Z5:17.635108:82.167863:1230.000000
0:3:0:1:Z1:23.365319:258.646227:2340.000000
0:4:0:2:Z3:5.798595:176.579702:1360.000000
0:5:0:2:Z5:28.269759:139.817949:1300.000000

●
●
●

0:74950:1:0:Z2:0.496621:136.855808:1940.000000
0:74951:1:1:Z2:12.262525:218.364958:1680.000000
0:74952:1:2:Z2:4.596897:83.888971:1800.000000
0:74953:1:2:Z4:33.366366:63.675139:930.000000
0:74954:1:0:Z3:41.240619:510.999933:1790.000000
0:74955:1:0:Z2:15.467429:106.908999:1320.000000
0:74956:2:2:NA:-1.000000:368.566870:0.000000

```

Figura 3.-Estructura archivo summary.txt

El archivo summary-NNN.txt contiene datos sobre el comportamiento de las personas en la simulación NNN. Posee una cabecera, seguida de tantas líneas como la cantidad de personas que se simularon. Ésta líneas tienen ocho campos, separados por el símbolo ‘:’. En la Figura 3 se muestra el contenido del archivo que corresponde a la simulación 000. En la Tabla 7 se detallan los campos y su respectiva descripción.

Tabla 7.- Descripción de los campos del archivo summary.txt

Campo	Descripción	Ejemplo
numExperiment	Número del experimento.	0 (Corresponde al código 000)
Id	Identificador de la persona simulada	15

Model	Identificador del modelo de la persona 0: Residente 1: Visitante tipo I 2: Visitante tipo II	0
groupoAge	Identificador del grupo etario de la persona 0: G1, 1: G2, 2: G3, 3: G4	2
safeZone	Identificador de la zona segura de la persona Z1, Z2, Z3, Z4, Z5: Identificadores de zonas seguras NA: la persona no tiene zona segura asignada	Z1
distanceToTargetPos	Distancia a la que quedo la persona de su objetivo inicial	13.871407 metros
responseTime	Número aleatorio que representa cuánto tiempo se demoró la persona en tomar la decisión de evacuar, desde que dieron la orden de evacuar. Si es -1, entonces la persona nunca pudo evacuar.	209.411742 segundos
evacTime	Tiempo que la persona se demoró en llegar a la zona de evacuación. Si es 0, entonces la persona nunca llegó a la zona de evacuación.	2300 segundos.

2.3.3 Archivo usePhone.txt

```
numExperiment:timeStamp usePhone
0:0:0
0:10:4333
0:20:4383
0:30:4261
0:40:4410
```

Figura 4.- Estructura archivo usePhone.txt

El archivo usePhone-NNN.txt contiene datos sobre el comportamiento de las personas en la simulación NNN, con respecto al uso de un teléfono móvil. Posee una cabecera y cada línea posterior, representa la cantidad de personas que utilizaron el teléfono en cierto instante de tiempo. Cada una tiene tres campos, separados por el símbolo ':'. En la Figura 4 se muestra el contenido del archivo que corresponde a la simulación 000. En la Tabla 8 se detallan los campos y su respectiva descripción.

Tabla 8.- Descripción de los campos del archivo usePhone.txt.

Campo	Descripción	Ejemplo
numExperiment	Número del experimento.	0 (Corresponde al código 000).
timeStamp	Tiempo de la medición.	60
usePhone	Cantidad de personas que utilizaron el teléfono móvil en el tiempo especificado.	4381

3 Diseño de la solución

Luego de analizar el problema se llega a lo siguiente: primero se requiere buscar y leer los archivos a procesar por el script de todas las simulaciones independiente de donde estén estos archivos, luego se requiere separar los datos por columnas, ya que cada dato de los archivos esta separado por dos puntos, se deben seleccionar solo los campos necesarios con los datos para realizar cada requerimiento, luego de hacer esto, los datos se redireccionarán a un archivo de texto temporal. Se deben leer estos datos y se deben ir procesando, para sacar el promedio, el mínimo y el máximo, luego de esto se irán redireccionando a un archivo de texto definitivo que ira guardando los datos de cada pregunta que se solicita. Al finalizar se eliminarán los archivos de texto temporales. A continuación, se especifica lo dicho anteriormente.

La búsqueda y lectura de datos serán tres **executionSummary-NNN.txt**, **summary-NNN.txt** y **usephone-NNN.TXT**, para la primera se requiere calcular el tiempo total de la simulación este tiempo esta definido como $\text{execMakeAgents} + \text{timeExecCal} + \text{timeExecSim}$. Esto se realizara mediante un arreglo y la información obtenida será redireccionada a un archivo temporal, para luego ser procesada y obtener el tiempo total de simulación, este tiempo será redireccionado a otro archivo temporal para luego realizar las estadísticas solicitadas, luego de eso la salida será redireccionada al archivo **metrics.txt**, luego para la memoria utilizada por el simulador se requiere la columna **maxMemory** de todas las simulaciones, mediante un arreglo se obtiene el valor y se redirecciona a un archivo temporal, luego de obtener ese valor para todas las simulaciones, se procesa ese archivo para obtener el promedio, máximo y mínimo y luego eso se redirecciona al archivo **metrics.txt**.

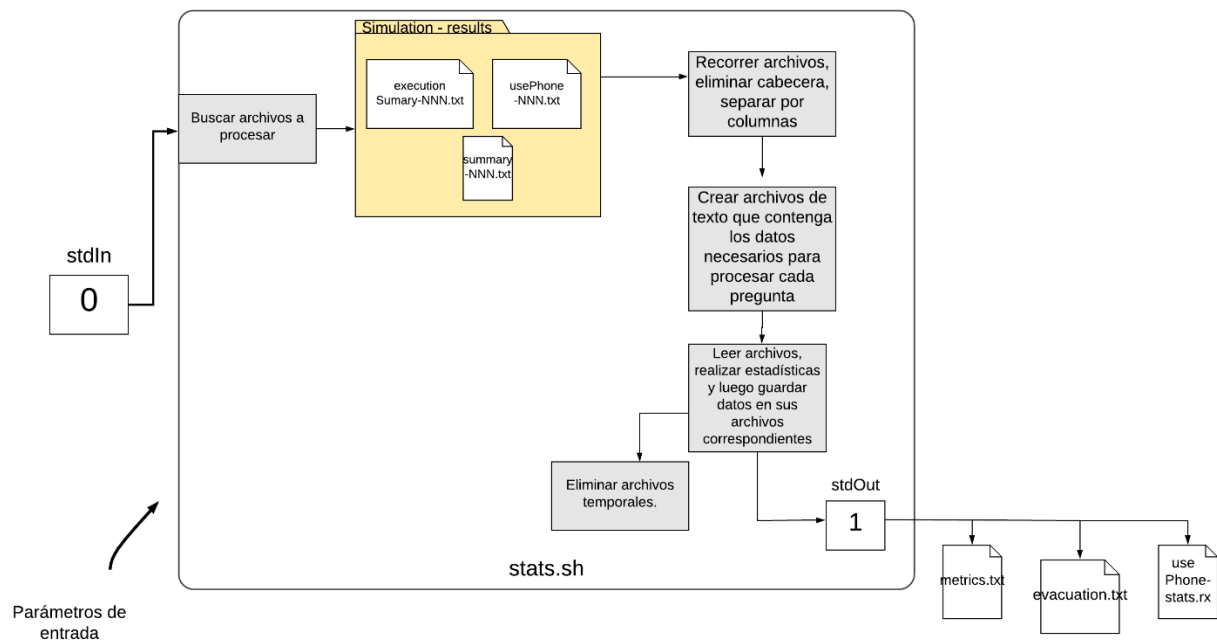
Para el segundo se requiere algo similar de lo realizado en el primero, se necesita obtener el valor de la columna **evacTime** de todas las simulaciones, pero aquí se tiene un par de condiciones ya que se tiene que diferenciar el grupo de personas, por ejemplo el residente-G0 del residente-G1, para hacer esto al recorrer el archivo se obtendrá solo lo que se requiere, en este caso es la columna 3,4,8, es decir **model**, **groupAge**, y **evacTime**, para obtener cada uno de los datos solicitados se recorrera el archivo y luego se crearan archivos temporales, para ser procesados y obtener el promedio, el mínimo y el máximo luego estos resultados serán redireccionados a un archivo llamado **evacuation.txt**.

Para el tercero, se hará lo mismo que en el uno y dos, lo que se necesita es el valor de la columna **usePhone**, por lo tanto, estos valores serán seleccionados y luego redireccionados a un archivo temporal, en este archivo temporal, en cada fila estarán los datos de cada simulación, por ejemplo en la fila 1 estarán todos los valores de la columna **usePhone** del archivo **usePhone-000.txt**, y así sucesivamente, después se contarán el número de líneas que tiene cada simulación, estas deberían ser 360, ya que en la línea 360 esta el ultimo instante de tiempo registrado en la simulación, así a través de un ciclo se ira calculando el promedio, min y max de cada instante

de tiempo de todas las simulaciones, luego la salida se irán redireccionado a un archivo llamado usePhone-stats.txt junto al instante de tiempo.

Cabe señalar que cuando se lee cada archivo se elimina la línea de cabecera tanto para la primera, segunda, y tercera, ya que esa línea no es necesaria para realizar las estadísticas. También luego de obtener las estadísticas en los archivos metrics.txt, evacuation.txt y usePhone-stats.txt, los archivos temporales creados serán eliminados. La solución a los problemas se puede visualizar en la figura 5.

Figura 5.- Diagrama del diseño de la solución



4 Metodología utilizada para la implementación

La implementación se llevo a cabo de la siguiente forma, primero se fue realizando cada parte por separado, es decir se creo un archivo con la parte 1, se comprobó que funcionara, y luego se paso a hacerlo mismo con la parte 2 y 3, finalmente se junto todo en el archivo principal stats.sh, se agregaron funciones a cada parte y una función para el modo de uso.

5 Pruebas de ejecución desde otros directorios

En la figura 6 se muestra una captura del script ejecutado desde el directorio /tmp de forma exitosa.

Figura 6.- Captura de ejecución script

```

sebastian@linux: /tmp
sebastian@linux:~$ cd /tmp/
sebastian@linux:/tmp$ pwd
/tmp
sebastian@linux:/tmp$ /home/sebastian/stats01/stats.sh -d /home/sebastian/taller-Sistemas-Operativos/
Directorio busqueda: /home/sebastian/taller-Sistemas-Operativos/
Obteniendo datos de: > /home/sebastian/taller-Sistemas-Operativos/UL/taller01/simulation-results/001/stats/executionSummary-001.txt
Obteniendo datos de: > /home/sebastian/taller-Sistemas-Operativos/UL/taller01/simulation-results/007/stats/executionSummary-007.txt
Obteniendo datos de: > /home/sebastian/taller-Sistemas-Operativos/UL/taller01/simulation-results/000/stats/executionSummary-000.txt
:
:
:
:

```


Ahora el script se ejecutará desde el directorio **/home**, el cual se ejecuta, pero con errores, en la figura 7 se puede apreciar lo que aparece.

Figura 7.- Captura ejecución script

```
sebastian@linux: /home
sebastian@linux: /home$ cd /home/
sebastian@linux: /home$ pwd
/home
sebastian@linux: /home$ /home/sebastian/stats01/stats.sh -d /home/sebastian/taller-Sistemas-Operativos/
Directorio busqueda: /home/sebastian/taller-Sistemas-Operativos/
/home/sebastian/stats01/stats.sh: line 61: metrics.txt: Permission denied
Obteniendo datos de: > /home/sebastian/taller-Sistemas-Operativos/U1/taller01/simulation-results/001/stats/executionSummary-001.txt
/home/sebastian/stats01/stats.sh: line 75: columnas.txt: Permission denied
/home/sebastian/stats01/stats.sh: line 77: sumaDeColumnas.txt: Permission denied
cat: columnas.txt: No such file or directory
/home/sebastian/stats01/stats.sh: line 83: maxMemory.txt: Permission denied
Obteniendo datos de: > /home/sebastian/taller-Sistemas-Operativos/U1/taller01/simulation-results/007/stats/executionSummary-007.txt
/home/sebastian/stats01/stats.sh: line 75: columnas.txt: Permission denied
/home/sebastian/stats01/stats.sh: line 77: sumaDeColumnas.txt: Permission denied
:
:
:
```

Luego se procede a ejecutar desde el directorio **/home** pero con el comando **sudo**, y se realiza de forma exitosa. Esto se puede apreciar en la figura 8.

Figura 8.- Captura ejecución script

```
sebastian@linux: /home
sebastian@linux: /home$ sudo /home/sebastian/stats01/stats.sh -d /home/sebastian/taller-Sistemas-Operativos/
Directorio busqueda: /home/sebastian/taller-Sistemas-Operativos/
Obteniendo datos de: > /home/sebastian/taller-Sistemas-Operativos/U1/taller01/simulation-results/001/stats/executionSummary-001.txt
Obteniendo datos de: > /home/sebastian/taller-Sistemas-Operativos/U1/taller01/simulation-results/007/stats/executionSummary-007.txt
Obteniendo datos de: > /home/sebastian/taller-Sistemas-Operativos/U1/taller01/simulation-results/000/stats/executionSummary-000.txt
:
:
:
```

6 Resultados.

A continuación, se presentarán los resultados luego de ejecutar el script **stats.sh**. En la figura 9, se muestra que se ejecuta el script de la siguiente forma: **./stats.sh** y se presenta por pantalla un mensaje que indica la forma correcta de como se debe ejecutar, se ingresa la forma correcta y el script comienza a ejecutarse.

Figura 9.- Captura de ejecución script

```
sebastian@linux: ~/stats01
sebastian@linux:~/stats01$ ./stats.sh
Uso: ./stats.sh -d <directorio datos> [-h]
    -d: directorio donde están los datos a procesar.
    -h: muestra este mensaje y termina.
sebastian@linux:~/stats01$ ./stats.sh -d /home/sebastian/taller-Sistemas-Operativos/
Directorio busqueda: /home/sebastian/taller-Sistemas-Operativos/
```

En la figura 10, se muestra los resultados obtenidos en el archivo **metrics.txt** con el comando **more**.

Figura 10.- Captura de resultados en archivo metrics.txt

```
sebastian@linux: ~/stats01
sebastian@linux:~/stats01$ more metrics.txt
tsimTotal:promedio:min:max
2687416:244310:235938:250040
memUsed:promedio:min:max
3187732:289793:288236:291784
```

En la figura 11, se muestran los resultados obtenidos en el archivo **evacuation.txt** con el comando `more`.

Figura 11.- Captura de resultados en archivo `evacuation.txt`.

```
sebastian@linux: ~/stats01
sebastian@linux:~/stats01$ more evacuation.txt
alls:promedio:min:max
1307782340:1585:0:3420
residents:promedio:min:max
1204398620:1586:0:3420
visitorsI:promedio:min:max
103383720:1597:0:3420
residents-G0:promedio:min:max
270935440:1673:0:3410
residents-G1:promedio:min:max
410947600:1481:0:3230
residents-G2:promedio:min:max
683000240:1694:0:3410
residents-G1:promedio:min:max
116091720:1640:0:3420
visitorsI-G0:promedio:min:max
22880200:1667:0:3400
visitorsI-G1:promedio:min:max
179515520:1460:0:3370
visitorsI-G2:promedio:min:max
135733740:1948:0:3390
visitorsI-G3:promedio:min:max
9727600:1554:0:3420
```

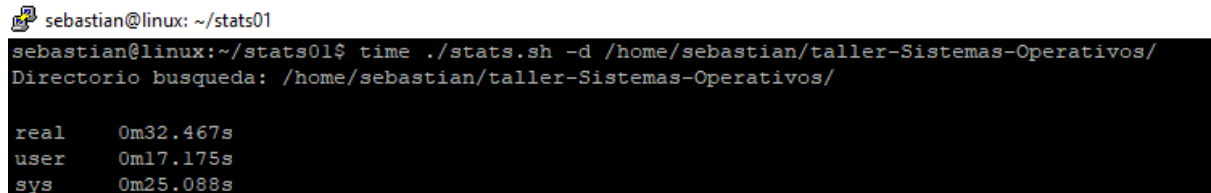
En la figura 12, se muestran parte de los resultados obtenidos en el archivo **usePhone-stats.txt** con el comando `more`.

Figura 12.- Captura de resultados en archivo `usePhone-stats.txt`

```
sebastian@linux: ~/stats01
sebastian@linux:~/stats01$ more usePhone-stats.txt
timestamp:promedio:min:max
1:4453.91:4530:4333
2:4428.55:4481:4369
3:4451.82:4585:4261
4:4439.91:4558:4372
5:4454.18:4529:4327
6:4415.64:4531:4297
:
:
:
353:158.545:190:139
354:162.091:185:139
355:164.636:187:147
356:156.909:174:142
357:157.091:176:139
358:161:185:147
359:160.545:183:138
360:152.545:175:130
```

En la figura 13 se muestra el tiempo de ejecución del script con el comando time.

Figura 13.- Captura del tiempo de ejecución del script



```
sebastian@linux: ~/stats01
sebastian@linux:~/stats01$ time ./stats.sh -d /home/sebastian/taller-Sistemas-Operativos/
Directorio busqueda: /home/sebastian/taller-Sistemas-Operativos/

real    0m32.467s
user    0m17.175s
sys     0m25.088s
```

Se obtienen tres resultados: real, user, sys, el primero se refiere al tiempo transcurrido entre la ejecución (desde que se presiona enter) y la finalización del proceso, el segundo es la cantidad de tiempo de CPU gastado en código de modo usuario (fuera del núcleo) durante la ejecución del proceso y el tercero es la cantidad de tiempo de CPU que transcurre en el núcleo al ejecutar el proceso.

7 Conclusión.

Como conclusión se puede decir que se cumplió con el objetivo, para esto había que tener muy claro cómo se debía abordar el problema, y gracias a un diseño previo se pudo llevar a cabo de mejor forma la solución. Algo a destacar es que se necesitó de varias horas de investigación para encontrar los comandos que hicieran determinada acción y también de varias horas de práctica en la implementación.

Se aprendió algo nuevo, del cual no se tenía conocimiento y que puede ser una herramienta muy utilizada para procesar archivos de texto y cosas relacionadas.

8 Referencias

- [1] “Bash”, *Wikipedia*, 2020. [En Línea]. Disponible en <https://es.wikipedia.org/wiki/Bash>. [Accedido: 27-jun-2020].
- [1] “The UNIX® Standard”, *opengroup*. [En Línea]. Disponible en: <https://www.opengroup.org/membership/forums/platform/unix>. [Accedido: 30-jun-2020]