Module code and Module Name

CS5001NA Networks and Operating Systems

Assessment Weightage & Type

20% Individual Coursework

Year and Semester 2019-20 Autumn

Student Name: Ishan Gurung

London Met ID: 19031315

College ID: NP01NT4A190139

Assignment Due Date: April 11, 2021

Assignment Submission Date: April 11, 2021

Word count (Task B):2014

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Acknowledgement

I would like to thank my teachers, Puranjan Acharya and Mr. dipeshwor for offering this course, which greatly helps me in understanding the Unix system. Mr. Umesh will make these courses understandable through appropriate guidance. He was so nice when he asked questions and he also helped me with a lot of research and learning new things. I am very grateful to him. I would like to thank my friends who helped me research and understand the questions and some of the commands and ultimately helped me complete the course in a limited time. This course has helped me a lot in developing my knowledge and skills. Thanks again for your support and it helped me finish the courses in advance.

Contents

1 Introduction

The Linux operating system is an open-source shell receives, processes, commands input, and executes it. This is the interface through which the user processes programs, commands and scripts. The shell is accessible from a terminal that runs it.

When you run the terminal, the Shell issues a command prompt(usually $) where we can enter the input you make when you press Enter. The output or result is then displayed on the terminal.

Enter a series of SHELL SCRIPTING commands to run the shell. You can combine long and repetitive scripts into one simple script that you can archive and run at any time. This reduces the effort required by a user. (guru99, 2020)

The instance provided for the script is to write a shell script that the program executes after the name and ID of the two parameters of the program are specified. Then, when a user is asked to enter a secret key and if the user enters the correct secret key, the program greets the user by entering their name and ID and the date and time of execution, otherwise it asks for the key. enter four times and then exit the program. The program now shows the names and codes of the five countries that have played the football game and the user is asked to find out which country has the best football team. After the user enters the code, the program displays a description of the team, otherwise an error message is displayed and the user can guess until the correct option is selected. The program then displays the names of the five players who have played football with the corresponding codes and asks the user to enter three codes in the displayed list. The three code lists will then appear after the program and you will be asked to select one. If the selected code contains a readable file, a description of the reader is displayed, otherwise the corresponding error message is displayed. At the end, the user is asked if he will continue the program and if the user connects, the process described above will continue and if the program will not end by pressing y and n button.

Thus, the applied activity is implemented by building an interactive request that uses UNIX commands, user input for error checking, diagnosing these errors with the correct message, and testing the script.

2. Script

```
#!/bin/bash
#ishan Gurung
#This is PlayerRead function
playerRead(){

PS3="Choose any number form the list:- "
select Players in $p1 $p2 $p3
do
  if [[ -z "$Players" ]]
   then
        echo "Input is Invalid!"
   playerChoose #If input is invalid playerChoose function is called.
  else
        echo "You choose favourite player."
        if [[ $Players == KC ]];
        then
                        cat KC
        elif [[ $Players == LM ]];
        then
                cat LM
        elif [[ $Players == NJ ]];
        then
                cat NJ

        elif [[ $Players == ZZ ]];
        then
```

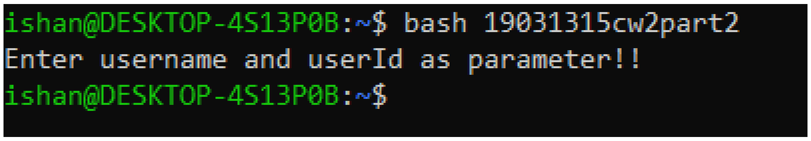
```
                echo -e "Player details is unavailable"
           countryList
        elif [[ $Players == HK ]];
        then
                echo -e "Player details is unavailable"
                countryList
        fi
        echo -e "\n Do you want restart the program (y|n)"

    read answer
    if [[ $answer == y ]]
    then
     countryList #countryChoose function is calling
    fi
 fi
 break
done
}

playerChoose(){
        echo -e "Choose 3 players using player code: \c"
        read p1 p2 p3

        if  [ "$p1" = "$p2" ] || [ "$p1" = "$p3" ] || [ "$p2" = "$p3" ]
        then
                echo "Player code is repeated! please check again!!"
                playerChoose
        fi
        case $p1 in
    "KC" | "LM" | "NJ" | "ZZ" | "HK")
  case $p2 in
        "KC" | "LM" | "NJ" | "ZZ" | "HK")
  case $p3 in
  "KC" | "LM" | "NJ" | "ZZ" | "HK")

     playerRead
     ;;
     *) echo "Third player invalid"
     playerChoose
     ;;
    esac
     ;;
     *) echo "Second player invalid"
     playerChoose
     ;;
    esac
     ;;
     *) echo "First player invalid"
      playerChoose
      ;;
    esac


}
playerList(){
        echo -e "Choose 3 Player using code:"
        echo -e "Player Name       Player Code"
        echo -e "Kiran Chemjong    KC"
        echo -e "Liinel Messi      KM"
        echo -e "Neymar Junior     NJ"
        echo -e "Zheng Zhi        ZZ"
        echo -e "Harry Kane       HK"
        playerChoose
}

countryChoose() {
        bestCountry="NEP"
        echo "Select country using country code"
        until [[ $countryChoosed == $bestCountry ]];
        do
                read -p "Choose the country code" countryChoosed
                if [[ $countryChoosed == $bestCountry ]];
                        then
                                break
                        else
                                echo "Enter the best country from list."
                        fi
                done
                 echo "You choose Nepal as the best country."
                 playerList
}

   countryList (){
        echo -e "Choose a country using the country code:"
        echo -e "County Name       Country Code"
        echo -e "Nepal            NEP"
        echo -e "China                      CHI"
        echo -e "Brazil            BRZ"
        echo -e "Argentina         ARG"
        echo -e "England           ENG"
```

```
  countryChoose #choose country function is calling
}

if [[ $# != 2 ]]
then
            echo "Enter username and userId as parameter!!"
    exit
fi

    name=$1
    id=$2
    count=0

    while [[ $count -lt 4 ]]
    do
      echo -e "Enter your Password:\c"
      read -s Password
      if [[ $Password == ishan ]];
      then
                  echo -e "\nWelcome To the Program." #Welcome
                  echo -e "id:  id" #display ID
                  echo -e "Username:  name" #display username
                  echo -e "Date of execution: $(date)" #display date of execution
                  countryList
                  break
      else
         echo -e "Wrong Password!"
      fi
        let count++ #value of count is increasing
```

done

Testing

3.1 Test 1



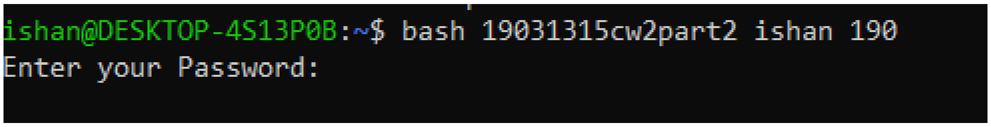
```
ishan@DESKTOP-4S13P0B:~$ bash 19031315cw2part2
Enter username and userId as parameter!!
ishan@DESKTOP-4S13P0B:~$
```

Figure 1 passing parameter while running program

Test no	1
Input	bash 19031315cw2part1
Expected Output	Program should not run and display the message
Actual Output	Program did not run and it shows message.
Result	Test was successful

Table 1 passing parameter while running program

3.2 Test 2



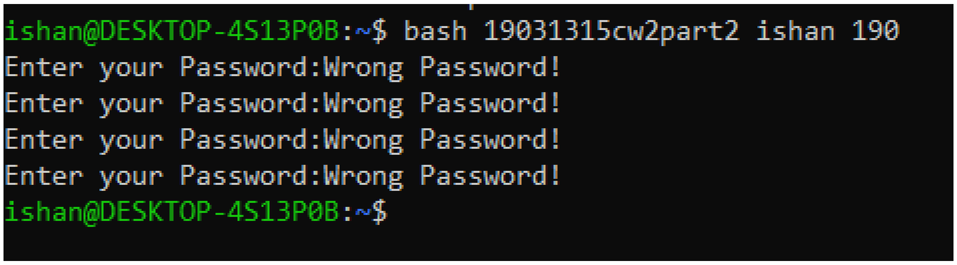
```
ishan@DESKTOP-4S13P0B:~$ bash 19031315cw2part2 ishan 190
Enter your Password:
```

Figure 2 passing valid parameter

Test no	2
Input	Bash 19031315cw2part2 ishan 190
Expected Output	Program should ask to enter the password
Actual Output	Program asked to enter the password
Result	Test was successful

Table 2 passing valid parameter

3.3 Test 3



```
ishan@DESKTOP-4S13P0B:~$ bash 19031315cw2part2 ishan 190
Enter your Password:Wrong Password!
Enter your Password:Wrong Password!
Enter your Password:Wrong Password!
Enter your Password:Wrong Password!
ishan@DESKTOP-4S13P0B:~$
```

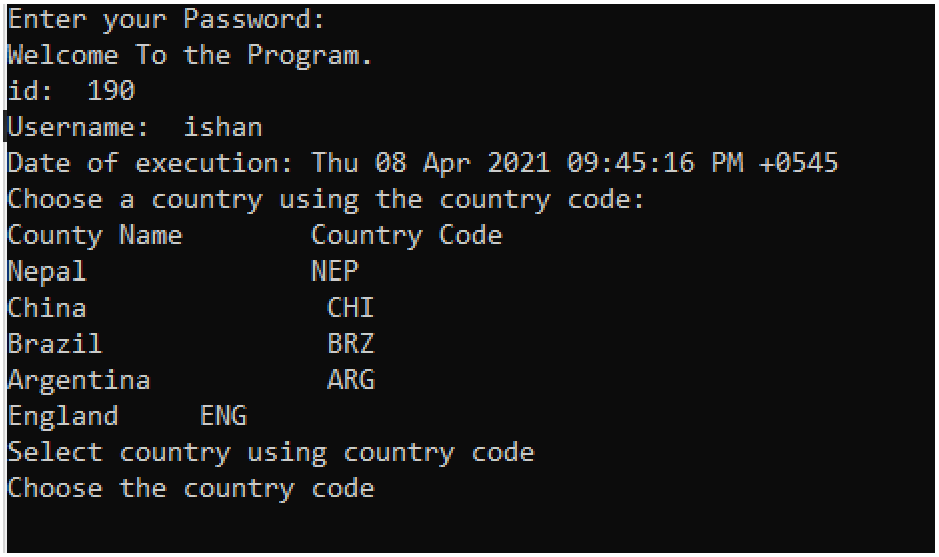
Figure 3entering invalid password

Test no	1
Input	111, 222, 233,444
Expected Output	Wrong password! Should be displayed for the four times and terminate the program

Actual Output	**Program displayed the Wrong password! Message for four times and terminate the program**
Result	**Test was successful**

Table 3 entering invalid password

3.4 Task 4



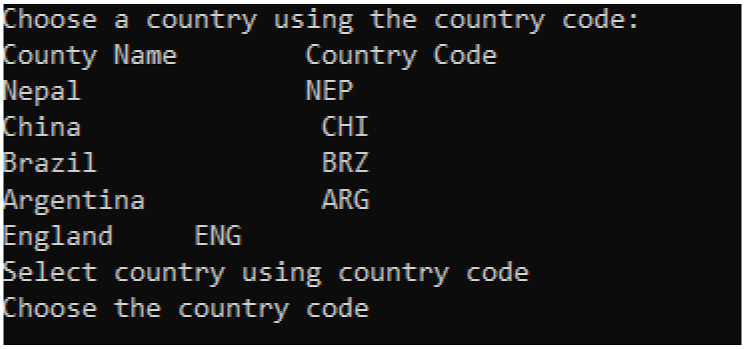
```
Enter your Password:
Welcome To the Program.
id:  190
Username:  ishan
Date of execution: Thu 08 Apr 2021 09:45:16 PM +0545
Choose a country using the country code:
County Name        Country Code
Nepal              NEP
China               CHI
Brazil              BRZ
Argentina           ARG
England     ENG
Select country using country code
Choose the country code
```

Figure 4entering valid password

Test no	**4**
Input	**Ishan**
Expected Output	**Program should start with the message and date of execution**
Actual Output	**Program started with the message and date of execution**
Result	**Test was successful**

Table 4 entering valid password

3.5 Task 5



```
Choose a country using the country code:
County Name        Country Code
Nepal              NEP
China               CHI
Brazil              BRZ
Argentina           ARG
England     ENG
Select country using country code
Choose the country code
```

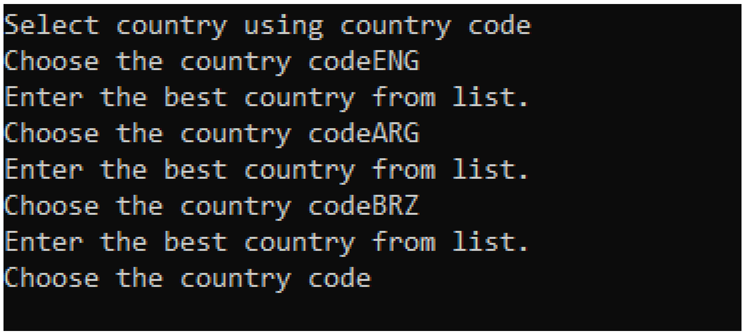
Figure 5 displaying country name

Test no	**5**

Input	**ishan**
Expected Output	**The program should display the country name and code in list**
Actual Output	**Program displayed the country name and code in the list**
Result	**Test was successful**

Table 5 displaying country name

3.6 test 6



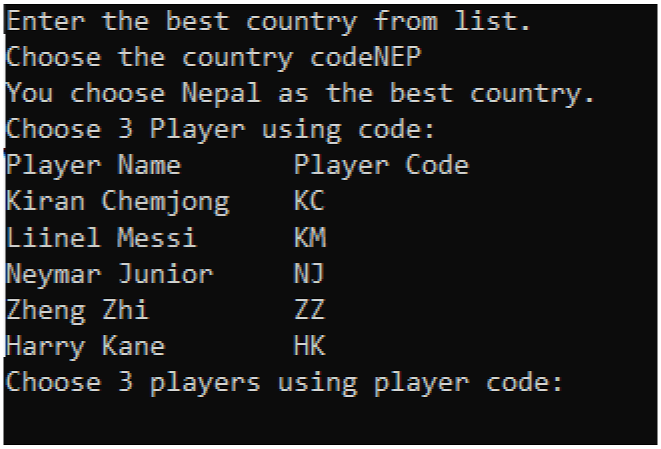
```
Select country using country code
Choose the country codeENG
Enter the best country from list.
Choose the country codeARG
Enter the best country from list.
Choose the country codeBRZ
Enter the best country from list.
Choose the country code
```

Figure 6 entering wrong country code

Test no	**6**
Input	**ENG, ARG, BRZ**
Expected Output	**The program should display the error message for choosing wrong country**
Actual Output	**The program displayed the error message**
Result	**Test was sucessful**

Table 6 entering wrong country code

3.7 test 7



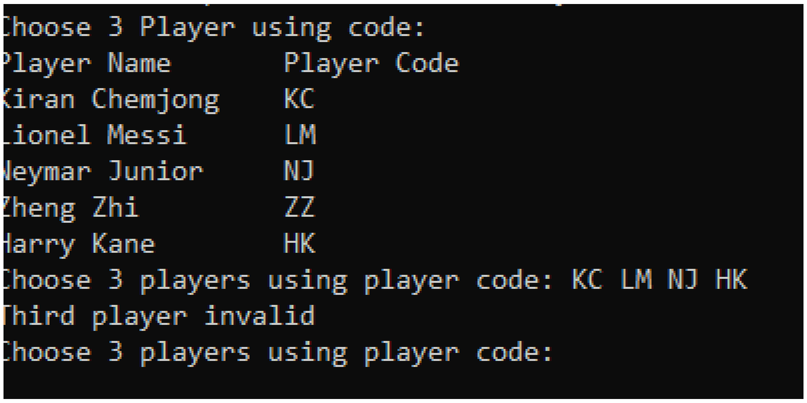
```
Enter the best country from list.
Choose the country codeNEP
You choose Nepal as the best country.
Choose 3 Player using code:
Player Name        Player Code
Kiran Chemjong     KC
Liinel Messi       KM
Neymar Junior      NJ
Zheng Zhi          ZZ
Harry Kane         HK
Choose 3 players using player code:
```

Figure 7 entering correct country name

Test no	7
Input	**NEP**
Expected Output	**The program should display the player's name along with player code.**
Actual Output	**Program displayed the player's name and player code**
Result	**Test was successful**

Table 7 entering correct country name

3.8 test 8



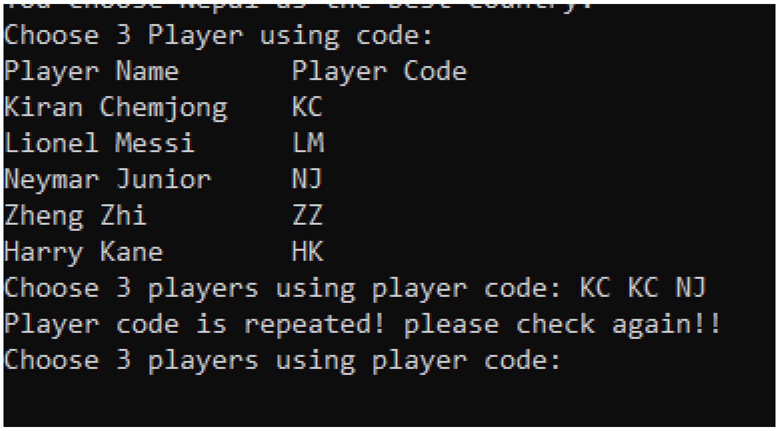
```
Choose 3 Player using code:
Player Name       Player Code
Kiran Chemjong    KC
Lionel Messi      LM
Neymar Junior     NJ
Zheng Zhi         ZZ
Harry Kane        HK
Choose 3 players using player code: KC LM NJ HK
Third player invalid
Choose 3 players using player code:
```

Figure 8 entering four player names

Test no	8
Input	**LC LM NJ HK**
Expected Output	**The program should display the message of invalid player**
Actual Output	**The program displayed the message of invalid player**
Result	**Test was successful**

Table 8 entering four country names

3.9 test 9



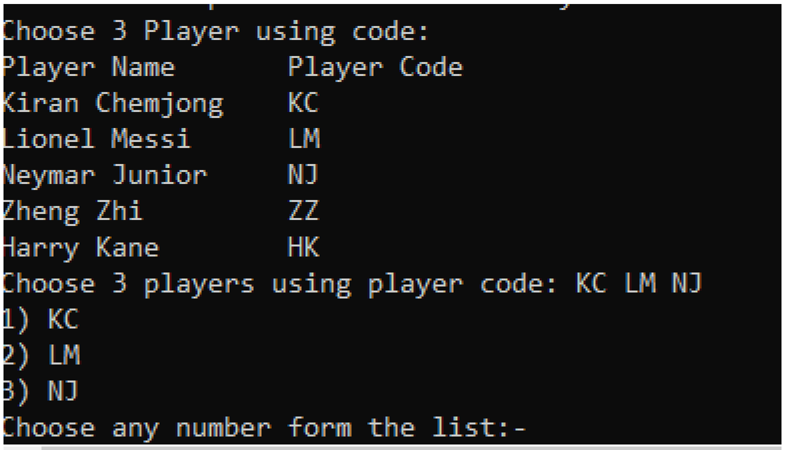
```
Choose 3 Player using code:
Player Name       Player Code
Kiran Chemjong    KC
Lionel Messi      LM
Neymar Junior     NJ
Zheng Zhi         ZZ
Harry Kane        HK
Choose 3 players using player code: KC KC NJ
Player code is repeated! please check again!!
Choose 3 players using player code:
```

Figure 9 entering same player code

Test no	**9**
Input	**KC KC NJ**
Expected Output	**The program should display the message of repeated player.**
Actual Output	**Program displayed the message of repeated player.**
Result	**Test was sucessful**

Table 9 entering same player code

3.10 test 10



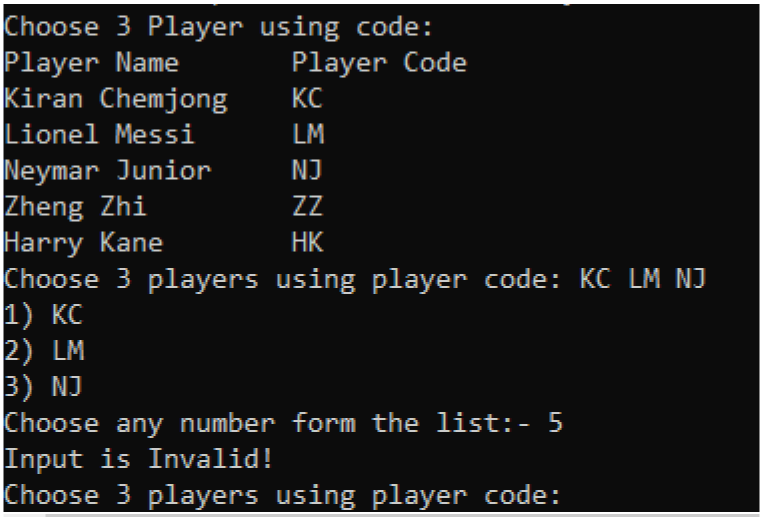
```
Choose 3 Player using code:
Player Name       Player Code
Kiran Chemjong    KC
Lionel Messi      LM
Neymar Junior     NJ
Zheng Zhi         ZZ
Harry Kane        HK
Choose 3 players using player code: KC LM NJ
1) KC
2) LM
3) NJ
Choose any number form the list:-
```

Figure 10entering valid player

Test no	**10**
Input	**KC LM NJ**
Expected Output	**Program should display the player in the list**
Actual Output	**Program displayed the chosen player in the list**
Result	**Test was successful**

Table 10 entering valid player

3.11 test 11



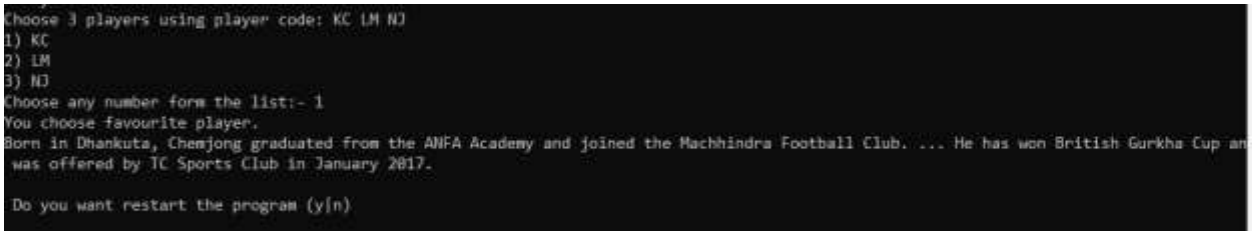
```
Choose 3 Player using code:
Player Name       Player Code
Kiran Chemjong    KC
Lionel Messi      LM
Neymar Junior     NJ
Zheng Zhi         ZZ
Harry Kane        HK
Choose 3 players using player code: KC LM NJ
1) KC
2) LM
3) NJ
Choose any number form the list:- 5
Input is Invalid!
Choose 3 players using player code:
```

Figure 11 entering invalid number from list

Test no	**11**
Input	**KC LM NJ**
Expected Output	**The program should display the appropriate message for entering the wrong input.**
Actual Output	**Program displayed the appropriate message for entering the wrong input**
Result	**Test was successful.**

Table 11 entering invalid number of the list

3.12 test 12



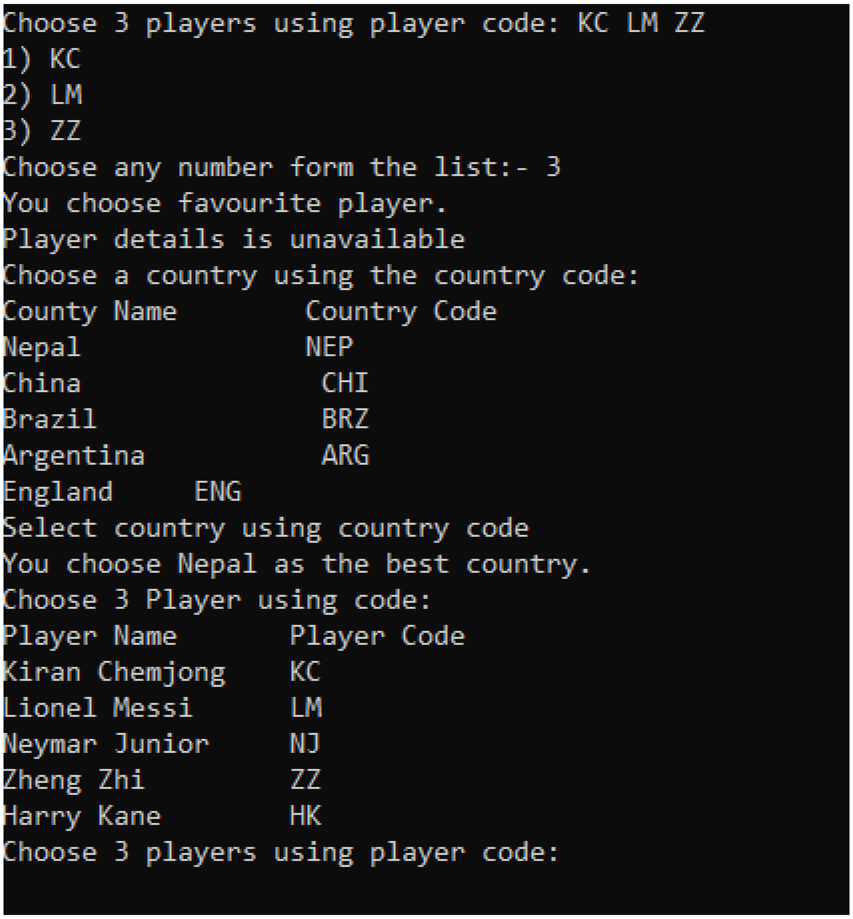
```
Choose 3 players using player code: KC LM NJ
1) KC
2) LM
3) NJ
Choose any number form the list:- 1
You choose favourite player.
Born in Dhankuta, Chemjong graduated from the ANFA Academy and joined the Machhindra Football Club. ... He has won British Gurkha Cup an
 was offered by TC Sports Club in January 2017.

 Do you want restart the program (y|n)
```

Figure 12 entering valid number from the list

Test no	**12**
Input	**1**
Expected Output	**The program should display the description of the chosen player list**
Actual Output	**The program displayed the description of the chosen player list.**
Result	**Test was successful**

Table 12 entering valid number from the list

3.13 test 13



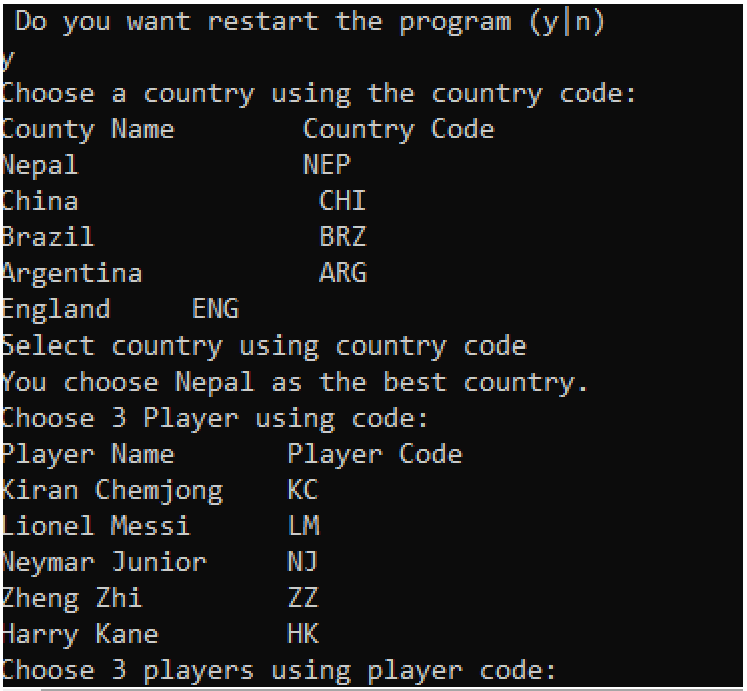
```
Choose 3 players using player code: KC LM ZZ
1) KC
2) LM
3) ZZ
Choose any number form the list:- 3
You choose favourite player.
Player details is unavailable
Choose a country using the country code:
County Name        Country Code
Nepal              NEP
China               CHI
Brazil              BRZ
Argentina           ARG
England     ENG
Select country using country code
You choose Nepal as the best country.
Choose 3 Player using code:
Player Name       Player Code
Kiran Chemjong    KC
Lionel Messi      LM
Neymar Junior     NJ
Zheng Zhi         ZZ
Harry Kane        HK
Choose 3 players using player code:
```

Figure 13 entering player number whose file doesn't exists

Test no	**13**
Input	**3**
Expected Output	**The program should display the message of unavailable details of player.**
Actual Output	**The program displayed the message of unavailable details of player**.
Result	**Test was successful.**

Table 13 entering player number whose file doesn't exists

3.14 test 14



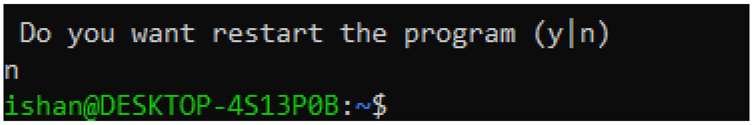
```
 Do you want restart the program (y|n)
y
Choose a country using the country code:
County Name          Country Code
Nepal                NEP
China                  CHI
Brazil                 BRZ
Argentina              ARG
England      ENG
Select country using country code
You choose Nepal as the best country.
Choose 3 Player using code:
Player Name        Player Code
Kiran Chemjong     KC
Lionel Messi       LM
Neymar Junior      NJ
Zheng Zhi          ZZ
Harry Kane         HK
Choose 3 players using player code:
```

Figure 14 restarting the program

Test no	**14**
Input	**y**
Expected Output	**Program should run again**
Actual Output	**The program was started**
Result	**Test was successful.**

Table 14 restarting the program

3.15 Test 15



```
 Do you want restart the program (y|n)
n
ishan@DESKTOP-4S13P0B:~$
```

-Figure 15 terminating the program

Test no	**15**
Input	**n**
Expected Output	**Program should terminate**
Actual Output	**Program was terminated**
Result	**Test was successful**

Table 15 terminating the program

4. Descriptions of three files

4.1 KC

Born in Dhankuta, Chemjong graduated from the ANFA Academy and joined the Machhindra Football Club. ... He has won British Gurkha Cup and Aaha Gold Cup Football Tournament. After being the best goalkeeper of south Asia of the time, Kiran was offered by TC Sports Club in January 2017.

4.2 LM

Lionel Messi was born, 24 June 1987, in Rosario, Argentina to a working-class family. His father was a factory steel worker, and his mother a cleaner.

4.3 NJ

Neymar da Silva Santos Júnior was born 5 February 1992), known as Neymar, is a Brazilian professional footballer who plays as a forward for Ligue 1 club Paris Saint-Germain and the Brazil national team. He is widely regarded as one of the best players in the world.

5. Conclusion

The first part of the course covered the relevant UNIX commands, which introduce a small interactive program developed using the UNIX command and the scripting language, e.g. bash shell script, in a well-known UNIX shell environment. Thanks to the whole course with the Linux operating system, which was run under a virtual machine. Linux is an open-source software operating system that runs on different platforms. All commands were executed under the Linux shell terminal.

The script added to the script had to write a shell script that was run based on that scenario. Therefore, the task was to create an interactive request using UNIX commands, check user input for errors, diagnose those errors with an appropriate message and test the script.

Thanks to this course, I have improved my knowledge and understanding of the UNIX command and the Ubuntu shell terminal. In addition, this course helped create the basics and concepts of the bash shell scripts. However, presenting a small interactive program using scripting language was not an easy task. However, this course inspired me to learn more about UNIX-based operating systems and bash shell scripts.

Task B

Introduction

In the task B of coursework part2, it is to research about the network operating system focus on memory issues or management of network operating system such as UNIX, which consists of the technical report that focus on the different reference site of the memory management. By the different technical link, it learns about the different memory management techniques. It provides a detailed description of types of the memory management.

Memory management is a function of an operating system in which it handles about main memory and moves processes back and forth between main memory and the disk while it is implementation. The management of memory keeps on tracking of each and every memory location in which it allocated to the process or it may be free. In each times Management checks the much memory allocated to process. It processes to decide in which it get memory at what particular time.

Each time memory management checks the free or unallocated part of the memory and updates the status accordingly. (Anon., 2018)

1.1.1 Aims

In task B, it focuses on examining and determining the course of investigation of memory management in the operating system in which it describes about the types of memory and the dynamic allocation of memory, the memory hierarchy and the reference position. The main aim of this report is to present a technical report aimed at the reference site in the field of memory management. In the operating system it is also known as computer architecture, for its example memory management and its hierarchy, a cache memory edition algorithm.

1.1.2 Objectives

Objectives of technical report are:

- Research and self-knowledge from the various sources, such as books, magazines, articles, online sites and studies of the memory management of operating system.

- Practice a good time management for the successful project completion.

2.1 Background of memory management

Memory consists of a wide array of word and action, in which its own address. The CPU fetches instructions from the memory based on the value of the program counter. These counter leads to additional to loading from a saving to certain memory address.

Usually, a program found a disk as a binary executable file. The program must be remembered and put in the process for it to be executed. Depending on the memory management in use, during its execution it is process to move between disk and memory. The input queue is the process collection on disk that is waiting to run the memory. (Anon., 2019)

2.1.1 Hierarchy of memory

Although the main distinction between main / auxiliary is widely useful, organization of computer memory creates a hierarchy of levels, and arranged from the very small, fast, expensive CPU program hierarchy to small, fast cache memory; larger DRAM; very large hard disks; and slow and cheap nonvolatile backup storage. Due to the speed of the modern hard disks and same location as the reference property, funds can function properly and virtual memory can be implemented. (Anon., 2018)

The five memory hierarchies are registers, cache, main memory, magnetic disks and magnetic tapes. The first three hierarchies are volatile memory, which means that if you have no energy, you will automatically lose your saved data. Although the last two hierarchies are not volatile, this means that the data is stored permanently.

A memory item is a collection of storage devices that store binary data in bit format. Generally, memory can be divided into two categories, such as volatile and non-volatile. (Elprocus, 2013)

2.1.2 Hierarchy of memory management

When designing a computer system, Memory hierarchy is an increase or improvement of that organizes memory so as to reduce access times. The memory hierarchy was developed based on a behavior of the program called the district of references. The following figure clearly shows the different levels of the memory hierarchy : (Rishabhjain, 2018)

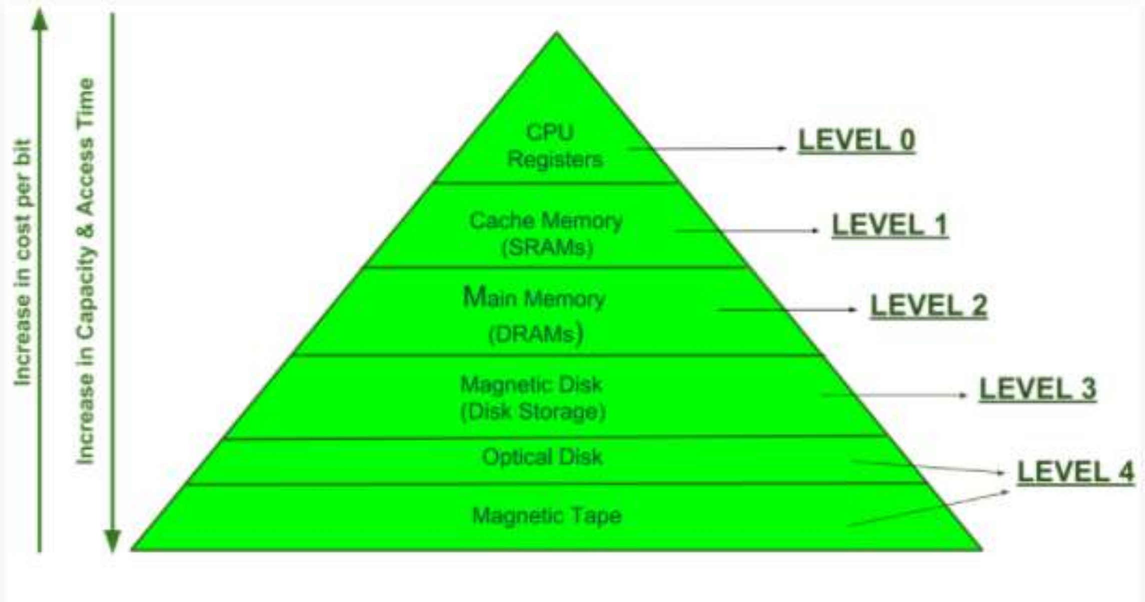


Figure 16 Arrow diagram for memory management

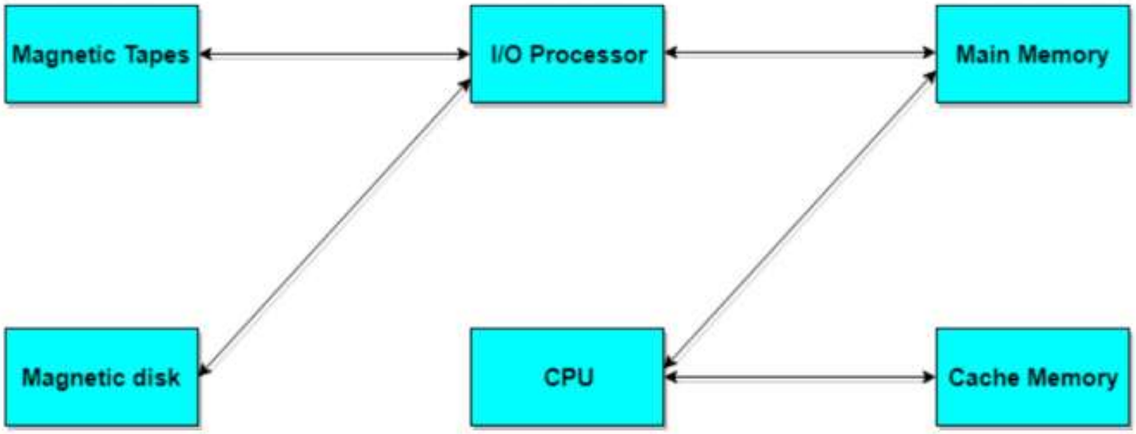


Figure 17 Arrow diagram for memory management

The five types of memory storage are discussed below:

1. Register:

Usually, The program in the computer processor register is a static RAM or SRAM, usually the processor of the computer which is used holding or to store 64 or 128 bit data words. The program counter is the most important and can be found in all processors. Computers, such as computers with complex instruction sets computer, usually have the same number of programs to register for accepting main memory, while computers with reduced RISC instruction sets have multiple programs to register.

2. Cache Memory

Main memory data or content, which is often used by the CPU, is cached so that in the short period of time the processor can easily access this data. If the CPU needs to access memory, it first checks the required data in the cache. When cached, data is read from fast memory. Otherwise, the CPU moves in main memory for the requried data. (Anon., 2018)

3. Main Memory

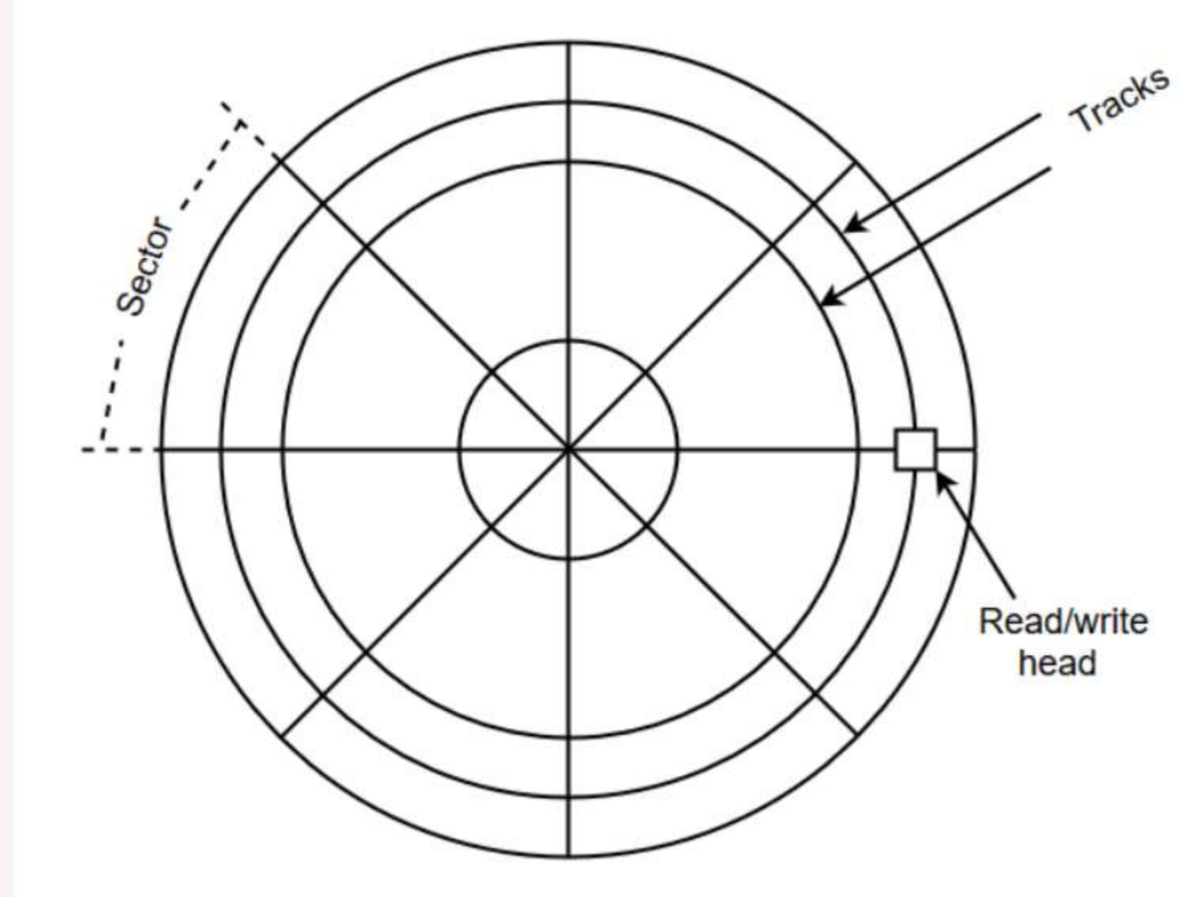
Memory that points directly to the CPU, auxiliary memory, and cache memory is called main memory. Most of the memory is made up of RAM and ROM and most of the RAM chips.

RAM: Random access memory Dram: Dynamic RAM made of capacitors and transistors and must be updated every 10 ~ 100 ms. It is slower and cheaper than SRAM. NVRAM: Non-volatile RAM, keeps your data even when it is turned off. Example: flash memory. (Anon., 2018)

ROM Read-only, non-volatile memory and permanent storage location other than information. Some commonly used ROMs are PROM (programmable ROM), EPROM (erasable PROM PROM) and EEPROM (erasable PROM electricity).

4. Magnetic Disks

It is a type of memory of magnetic computer disk which is constructed circular plastic disks, otherwise metal, with magnetic material. Mostly Two disk surfaces are often used and many discs can be stacked on a spindle using the read or write heads operation in which several disks which stached on one. (Anon., 2018)



Magnetic disks

- The memory bits are stored in points along the concentric circles called bands in the magnetized surface.

- The circles (bands) are generally divided into sections, known sectors.

5. Magnetic Tape

It is a storage medium which allows the archiving, collection and backup of data with a different variety of data. It is made of plastic strip coated with magnetic support. Magnetic tape drives can be stopped, started or moved back or restored. However, they cannot start or stop quickly enough between characters. Therefore, the information is recorded in fixed blocks referred as records. (Anon., 2018)

Locality of reference

In operating systems, the concept of the reference states that the operating system can only load the number of pages into the main memory which is free accessed by the CPU and operating system, rather than loading the whole process into the main memory. you can only load table page entries suitable for corresponding to multiple pages. (sing, 2019)

There are two kinds of locality of reference. They are:

- Temporal Locality
- Spatial Locality

Temporal Locality:

It means that Current data or instructions which is fetched may be needed soon. Therefore, these data or instructions should be stored in cache memory to avoid the same data in main memory which is known as temporal locality. If we refer to some data, there is a high probability that we will refer to them soon. (sing, 2019)

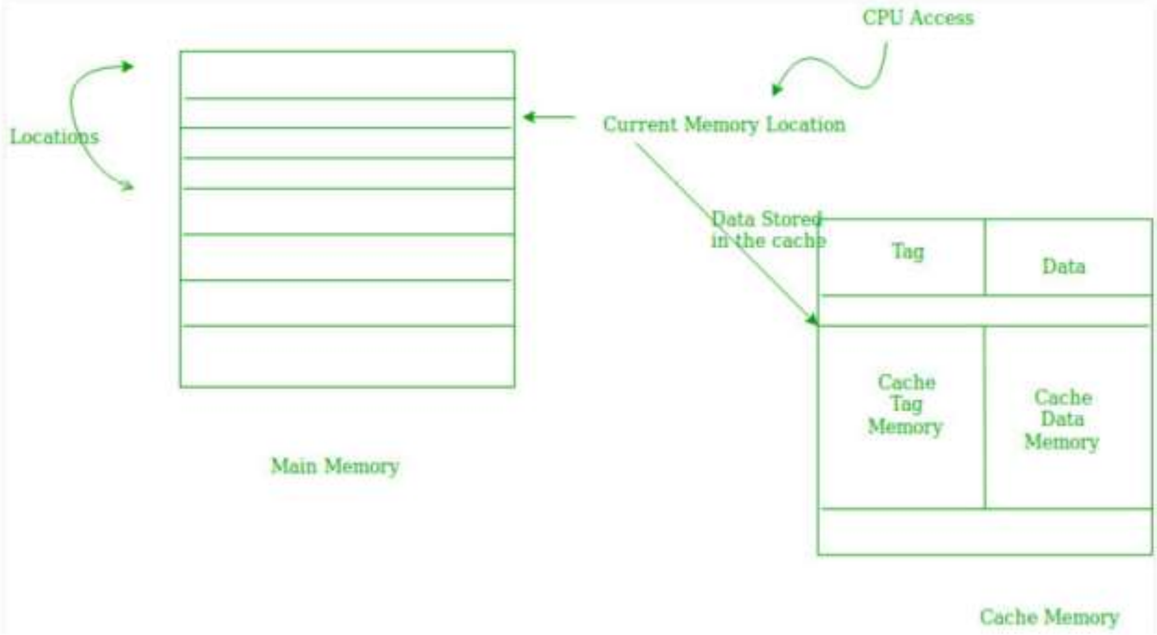


Figure 18 Temporal locality

Spatial Locality –

The spatial locality means that instructions or data which is close to the current memory location which is being fetched, which may be needed in the near future. This is a little different from the temporal locality. Here we are talking about memory locations that are almost present, while in space of time we are talking about the actual memory location which is being fetched. (sing, 2019)

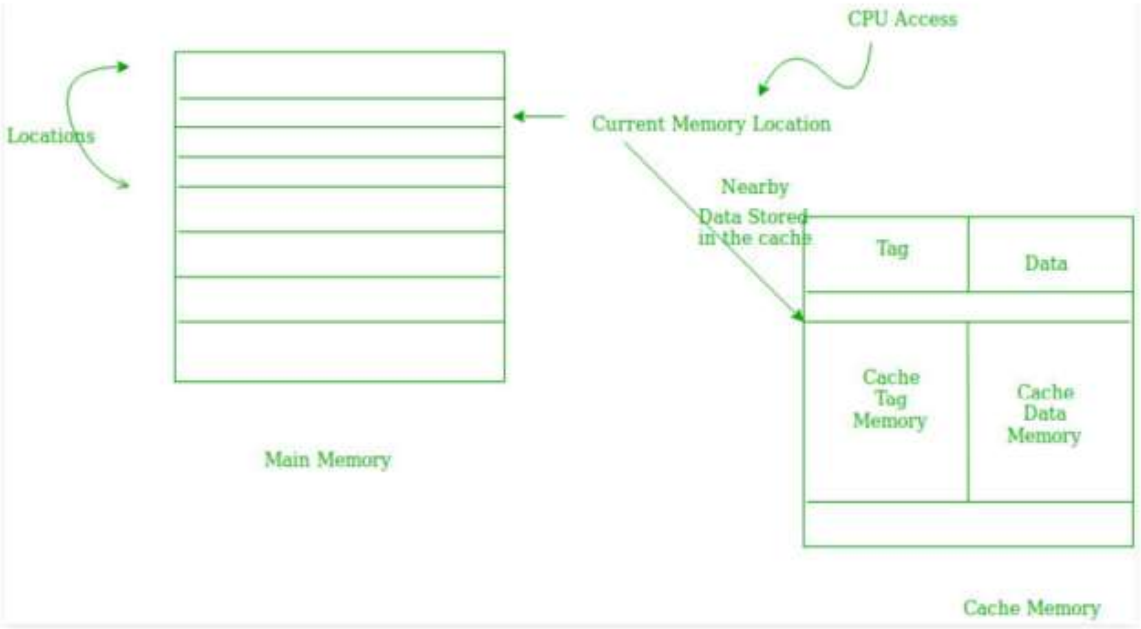


Figure 19 Spatial Locality

2.1.3 Memory allocation

It is primarily a computer hardware operation, but is managed through the operating system and software applications. The memory allocation process is very similar in the management of physical and virtual memory. When they are executed, program and service of Special memory are assigned to the requirements necessary for their implementation. When a program is completed, memory is freed and allocated to another program or merged with main memory. (janalta, 2018)

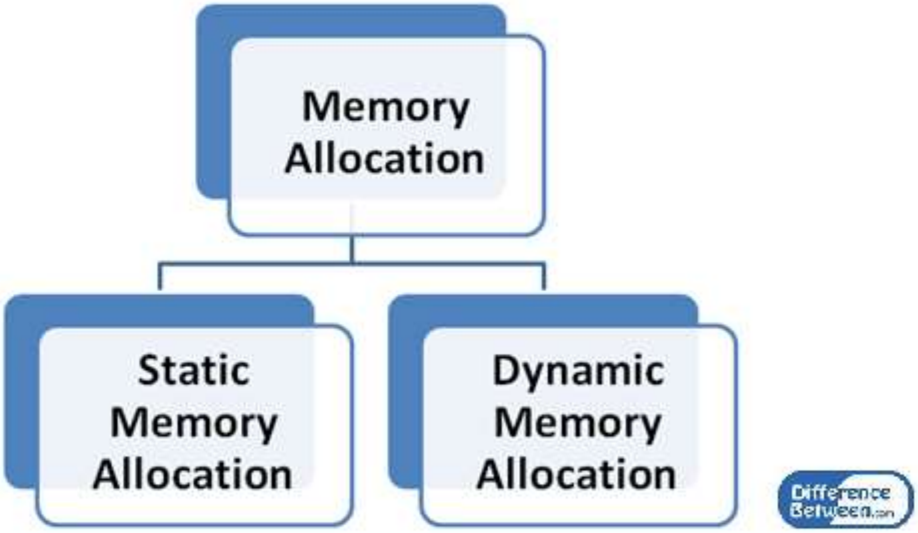


Figure 20 Memory allocation

There are two core types of memory allocation;

Static memory allocation:

The program allocates memory during compile time.

The advantage of static allocation is:

- The process effective way to allocating static memory is allocating memory.
- Before the execution starts all memory, allocation operation is performed.

Dynamic memory allocation:

The programs run with the memory at runtime. (Techopedia, 2020)

The advantage of dynamic allocation is:

- Dynamic memory allocation is allocation which allows flexible allocation of memory process.
- Dynamic memory allocation is allocation wile the program is running it reduces memory waste by allocating memory.

3. Dynamic Storage allocation

Dynamic memory allocation is a manual allocation and freeing of memory distributed in the current period of the program needs and is a method of dividing the ownership of limited memory resources between many data and codes. A dynamically allocated object remains occupied until it is explicitly distributed, by the programmer or the garbage collector; this is significantly different from automatic and static memory allocation. The life cycle of such an object is said to be dynamic. Out of one of the holes a new dynamic request for memory might return a range of the address. (QtSpim, 2015)

4. Description

4.1 Large physical memory

Physical memory refers to the RAM of the system, the form of cards (DIMMs) which is usually attached onto the motherboard. Which is also called as a primary memory. Physical memory is the only storage type which is directly accessibly to the CPU which execute to holds to the instruction of the program. In the linear fashion physical memory is linear addressable, memory address increases and each and ever byte is directly addressable. In the physical memory virtual memory adds a layer of abstraction and its memory offers many benefits such as the ability to maintain separate address spaces and the it has the ability to use the physical memory as a large cache of the physical disk. (Anon., 2014)

4.2 Memory placement

4.3 Cache coloring

In the computer science, cache coloring is also known as a page coloring which is the process to attempt the to allocate free pages which is contiguous of CPU caches in point of view. By which it maximizes the total number of the page cached by the processor. In the operating system cache coloring is typically employed by low-level dynamic memory allocation code. When it is mapping virtual to physical memory. (Anderson, 2007) (Britannica, 2017)

4.4 Page size variation

4.5 Paging

Paging is a storing mechanism which allows the operating system to retrieve processes from secondary storage to primary storage as a form of a page. In paging mode, the main memory is divided into small and medium-sized blocks of physical addresses called frames. Frame sizes should match the page size to maximize the utilization of the main memory or primary memory usage and avoid external fragmentation. Paging is used to access data faster and this is a logical concept. (Guru99, 2021)

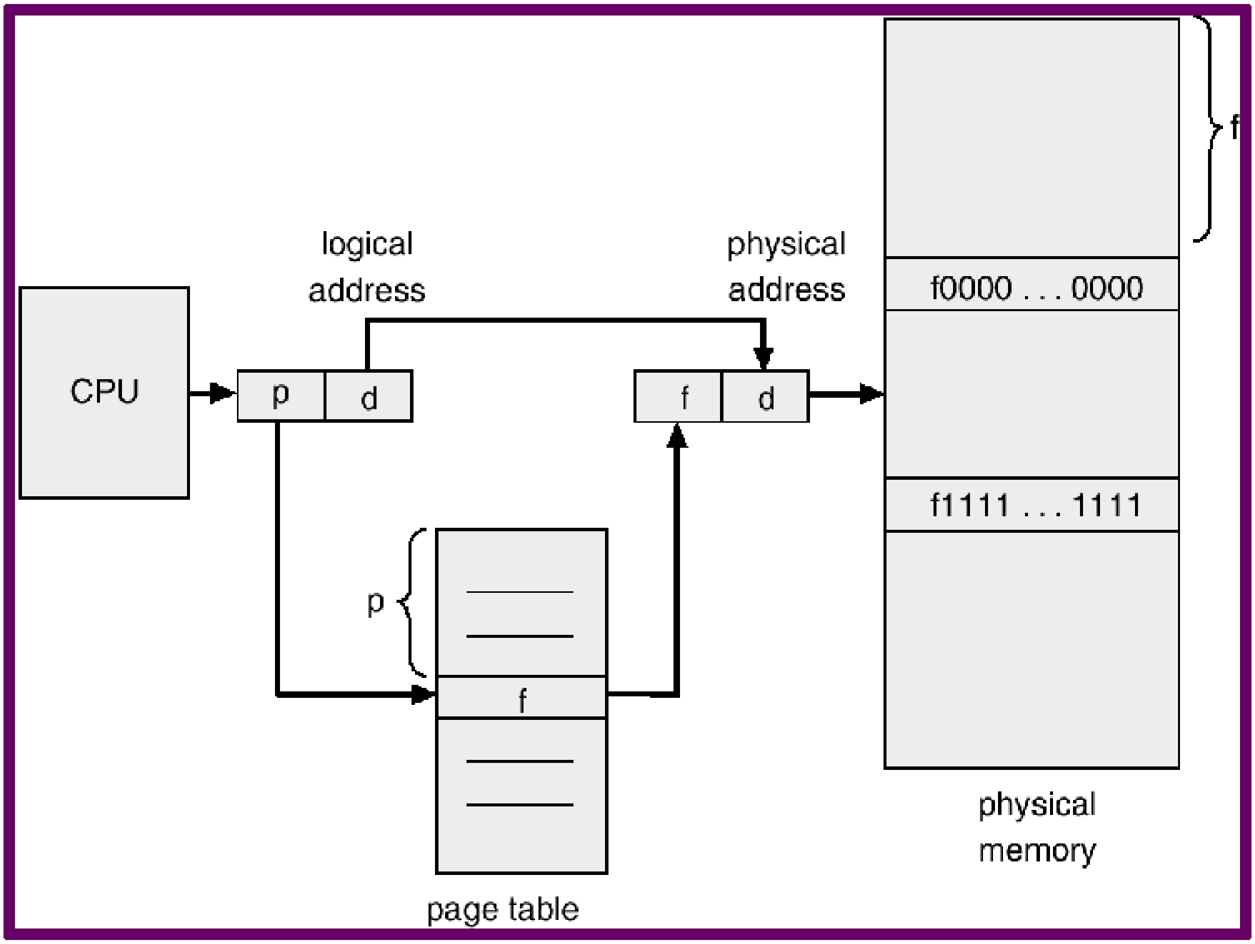


Figure 21 Paging

5. Conclusion

In coursework part 2, task of course B was to research the memory management of Network operating systems, which consists of a technical report that focuses mainly on the memory issues. So, I came to the conclusion that the technical part of this task depends on research. Therefore, the collection of information from the previous report on the reference site and the memory issues helped to generate some knowledge on memory management.

The difficulties that arise when I analyzed the technical meaning of the memory issues and reference space were not an easy task. I did a lot of high-quality research on various magazines and books to get through the project. I have collected a lot of information about page coloring and memory management. It was very informative for me. Now I understand the concept of memory management. In summary, this project inspired me to learn more about operating system memory management.

References

Anderson, j. M., 2007. *wikipedia.* [Online]
Available at:
https://en.wikipedia.org/wiki/Cache_coloring#:~:text=In%20computer%20science%2C%20cache%20coloring,pages%20cached%20by%20the%20processor.

Anon., 2014. *wikibooks.* [Online]
Available at: https://en.wikibooks.org/wiki/Operating_System_Design/Physical_Memory

Anon., 2018. *Britannica.* [Online]
Available at: https://www.britannica.com/technology/computer-memory/Memory-hierarchy

Anon., 2018. *java point.* [Online]
Available at: https://www.javatpoint.com/coa-auxiliary-memory

Anon., 2018. *javapoint.* [Online]
Available at: https://www.javatpoint.com/coa-memory-hierarchy

Anon., 2018. *studytoinight.* [Online]
Available at: https://www.studytonight.com/computer-architecture/memory-organization
[Accessed 2021].

Anon., 2018. *tutorialpoint.* [Online]
Available at: https://www.tutorialspoint.com/operating_system/os_memory_management.htm

Anon., 2019. *cs.iit.edu.* [Online]
Available at:
http://www.cs.iit.edu/~cs561/cs351/memory_management_FS_sched/index_mm_01.html#:~:text=Memory%20consists%20of%20a%20large,value%20of%20the%20program%20counter.&text=Depending%20on%20the%20memory%20management,and%20memory%20during%20its%20execution.

Britannica, 2017. *Britannica.* [Online]
Available at: https://www.britannica.com/technology/cache-memory

Elprocus, 2013. *Elprocus.* [Online]
Available at: https://www.elprocus.com/memory-hierarchy-in-computer-architecture/

Guru99, 2021. *Guru99.* [Online]
Available at: https://www.guru99.com/paging-in-operating-system.html

janalta, 2018. *techopedia.* [Online]
Available at: https://www.techopedia.com/definition/27492/memory-allocation

QtSpim, 2015. *chortle.ccsu.edu.* [Online]
Available at: https://chortle.ccsu.edu/assemblytutorial/chapter-33/ass33_3.html#:~:text=Dynamic%20memory%20allocation%20is%20when,a%20block%20of%20main%20memory.&text=The%20memory%20comes%20from%20above,return%20previously%20dynamically%20allocated%20memory.
[Accessed August 2015].

Rishabhjain, 2018. *geeksforgeeks.* [Online]
Available at: https://www.geeksforgeeks.org/memory-hierarchy-design-and-its-characteristics/

sing, b., 2019. *geeksforgeeks.* [Online]
Available at: https://www.geeksforgeeks.org/locality-of-reference-and-cache-operation-in-cache-memory/

Bibliography

Anderson, j. M., 2007. *wikipedia.* [Online]
Available at:
https://en.wikipedia.org/wiki/Cache_coloring#:~:text=In%20computer%20science%2C%20cache%20coloring,pages%20cached%20by%20the%20processor.

Anon., 2014. *wikibooks.* [Online]
Available at: https://en.wikibooks.org/wiki/Operating_System_Design/Physical_Memory

Anon., 2018. *Britannica.* [Online]
Available at: https://www.britannica.com/technology/computer-memory/Memory-hierarchy

Anon., 2018. *java point.* [Online]
Available at: https://www.javatpoint.com/coa-auxiliary-memory

Anon., 2018. *javapoint.* [Online]
Available at: https://www.javatpoint.com/coa-memory-hierarchy

Anon., 2018. *studytoinight.* [Online]
Available at: https://www.studytonight.com/computer-architecture/memory-organization
[Accessed 2021].

Anon., 2018. *tutorialpoint.* [Online]
Available at: https://www.tutorialspoint.com/operating_system/os_memory_management.htm

Anon., 2019. *cs.iit.edu.* [Online]
Available at:
http://www.cs.iit.edu/~cs561/cs351/memory_management_FS_sched/index_mm_01.html#:~:text=Me

mory%20consists%20of%20a%20large,value%20of%20the%20program%20counter.&text=Depending%20on%20the%20memory%20management,and%20memory%20during%20its%20execution.

Britannica, 2017. *Britannica.* [Online]
Available at: https://www.britannica.com/technology/cache-memory

Elprocus, 2013. *Elprocus.* [Online]
Available at: https://www.elprocus.com/memory-hierarchy-in-computer-architecture/

Guru99, 2021. *Guru99.* [Online]
Available at: https://www.guru99.com/paging-in-operating-system.html

janalta, 2018. *techopedia.* [Online]
Available at: https://www.techopedia.com/definition/27492/memory-allocation

QtSpim, 2015. *chortle.ccsu.edu.* [Online]
Available at: https://chortle.ccsu.edu/assemblytutorial/chapter-33/ass33_3.html#:~:text=Dynamic%20memory%20allocation%20is%20when,a%20block%20of%20main%20memory.&text=The%20memory%20comes%20from%20above,return%20previously%20dynamically%20allocated%20memory.
[Accessed August 2015].

Rishabhjain, 2018. *geeksforgeeks.* [Online]
Available at: https://www.geeksforgeeks.org/memory-hierarchy-design-and-its-characteristics/

sing, b., 2019. *geeksforgeeks.* [Online]
Available at: https://www.geeksforgeeks.org/locality-of-reference-and-cache-operation-in-cache-memory/