LONDON
METROPOLITAN
UNIVERSITY

ITAHARI
INTERNATIONAL
COLLEGE

Module Code & Module Title

CS5001NA Network and Operating System

Assessment Weightage & Type

20% Individual Coursework

Year and Semester

2019-20 Autumn / 2019-20 Spring

Student Name: Girija Tamang

London Met ID: 18030995

College ID: NP05CP4S190007

Assignment Due Date: 20 Mar 2020

Assignment Submission Date: 09 April 2020

Submitted To: Pratik Karki

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.

Table of Contents

Task A 1

1. Introduction 1
2. Script 2
3. Testing 12
4. Content of three files 21
5. Conclusion 23

Task B 24

1. Introduction 24
2. Aim and Objective 24
3. Background 25
4. Conclusion 33
5. References 34

Tables of Figures

Figure 1:Running Script file without username and id 12
Figure 2: Running Script file with username and id 12
Figure 3: Entering wrong password for five times. 13
Figure 4: Entering right password in secret key. 14
Figure 5:Entering country name. 15
Figure 6:Choosing incorrect country code. 15
Figure 7: Giving correct country code.. 16
Figure 8: Selecting four players name. 17
Figure 9: Selecting same player code. 17
Figure 10: selecting wrong player id. 18
Figure 11: Selecting right player id. 18
Figure 12: Selecting player without having profile. 19
Figure 13: Continuing game. 20
Figure 14: Exiting game. 21
Figure 15: Hierarchy of memory management. 26
Figure 16: Locality of references 26
Figure 17:Temporal locality 27
Figure 18: Spatial locality. 28

Tables of Tables

Table 1: Running Script file without username and id 12

Table 2: Running Script file with username and id 12

Table 3: Entering wrong password for five times. 13

Table 4: Entering right password in secret key. 14

Table 5: Entering country name. 15

Table 6: Choosing incorrect country code. 15

Table 7: Giving correct country code. 16

Table 8: Selecting four players name. 17

Table 9: Selecting same player code. 17

Table 10: selecting wrong player id. 18

Table 11: Selecting right player id. 18

Table 12: Selecting player without having profile. 19

Table 13: Continuing game. 20

Table 14: Exiting game... 21

Task A

1. Introduction

The shell is a special user program that provides an interface for the software to access the operating system services. A shell is a software program which interprets the user's commands so that the operating system can understand them and perform the relevant functions. The user can type commands to execute functions such as running programs, opening and browsing files, and displaying processes that are currently running. The shell gets started when the user logs in or start the terminal. Shell is broadly classified into two categories. They are Command Line Shell and Graphical shell (Janssen & Janssen, 2017).

A shell script is a text file containing a sequence of commands for an operating system based on Unix. A shell script is generally created for sequences of commands in which a user wants to use it frequently to save time. In the DOS operating system, a shell script is called a batch file (geeksforgeeks.org, 2018).

In this coursework, we were assigned to build a script file developed in the environment of a well-known UNIX shell. This script is made for the user to implements interaction with the UNIX environment in a friendly manner which executes simple input/output operations. The main goals of the coursework are building user friendly script file using UNIX commands, making familiar about UNIX shell, executing simple input/output operation and error handling in script file. As per the scenario I have developed a user-friendly script file which executes simple input/output operations in a UNIX shell.

Aims and objectives of this coursework:

- Introduce UNIX shell and develop a script in the environment of UNIX Shell.
- Perform some simple input/output operations at the UNIX shell.
- Enhanced personal skills and knowledge in checking input errors and specifying errors with clear message.
- Using various command for creating and reading user inputs and handling errors.

2. Script

```
#!/bin/bash

#function Declare

Theme () {

 #This function contain welcome part.

        echo ""

        echo "·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:"

        echo "»»    WElCOME TO GAME OF CRICKET       ««"

        echo "·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:·:"

        echo ""

        echo "»» Name: Mr\Mrs " $1

        echo "»» Id:" $2

        echo "»» Date: " $( date )   #Print the current time.

        echo " "

}

CountryList () {

 #This function displays the country name with their respective code.

        echo "......................................"

        echo "           ROUND 1                  "

        echo "......................................"

        echo "-----------------------------------"
        echo "      Cricket Team            "
        echo "----------------------------------"
        echo "S.N.      Countries        Code"
        echo "»»»»       »»»»»»»»»»       »»»»"
        echo " 1        Australia         AUS"
        echo " 2        Bangladesh        BAN"
        echo " 3        Nepal             NEP"
        echo " 4        India             IND"
        echo " 5        England           ENG"
        echo ""
        echo "**********************************"
        SelectTeam #Here SelectTeam has been called.
}
SelectTeam () {
#This function is used for select team .
        echo ""
        echo "▸▸▸▸ Please guess the best Cricket Team from the above list"
        echo "  entering the country code?"
        teamcode=""
        until [[ $teamcode == IND ]] #Here, until loop used for condition check.
                do
                echo -e "»»»» Enter the Country code (Eg: NEP): \c"

                read teamcode #reads input.

                case $teamcode in   #Here, case statement is used to match the statement.

                AUS) echo "Sorry! Your guess is wrong. Please choose another Cricket Team."

          ;;

                BAN) echo "Sorry! Your guess is wrong. Please choose another Cricket Team."

          ;;

                NEP) echo "Sorry! Your guess is wrong. Please choose another Cricket Team."

          ;;

                IND) echo ""

                    echo "Congratulation! Your guess is absolutely Correct."

                    echo "India is the best team who has won the Cricket World Cup Twice."

                PlayerInfo

                ;;

                ENG) echo "Sorry! Your guess is wrong. Please choose another Cricket Team."

          ;;

                *) echo " Opps ! You have enter the wrong code or invalid input ! Please enter the available
```

```
code from the above list and choose the Best Cricket Team."

        esac

        done

        echo ""

}

PlayerInfo () {   #This function displays the player name with their respective code.

    echo "………………………………"

    echo "          ROUND 2               "

    echo "………………………………"

    echo "================================="

    echo "  Players of the Cricket Team     "

    echo "================================="

    echo " S.N.     Player        Code"

    echo " ~~~    ~~~~~~~~~        ~~~~"

    echo " 1     Paras Khadka      PK"

    echo " 2     Virat Kholi       VK"

    echo " 3     David Warner      DW"

    echo " 4     Ben Stokes        BS"

    echo " 5     Ross Taylor       RT"

    echo ""

    echo -e "»» Select three Players with their respective code: \c"
```

```
    read player   #Takes the input

    for pl in $player

    do

        if [[ $pl == "PK" || $pl == "VK" || $pl == "DW" || $pl == "BS" || $pl == "RT" ]]

        then

            continue

        else

            echo " Opps ! You have enter the wrong code or invalid input ! Please
enter the available

  code from the above list"

            echo ""

            PlayerInfo #calls the function itself, when the wrong input is provided

        fi

    done

    PlayerSelected $player  #choose parameter has been passed to PlayerChoosen
function

    }

    PlayerSelected() {

        if [[ $# == 3 && $1 != $2 && $1 != $3 && $3 != $2 ]] # Here if is used to check
whether the parameter is equal to 3 or not.

        then

            PS3="Select the best player with its respective number (like 1) : "
```

```
select playerid in $player
do
        case $playerid in
                "PK")
                        FileRead Paras #Here Paras file has been read.
                        break
                        ;;
                "VK")
                        FileRead Virat #Here Virat file has been read.
                        break
                        ;;
                "DW")
                        FileRead David #Here David file has been read.
                        break
                        ;;
                "BS")
                        FileRead Ben  #Here Ben file has been read.
                        break
                        ;;
                "RT")
                        FileRead Ross #Here Ross file has been read.
                        break
                        ;;
                esac
            done
            Restart  #restart has been called
        else
            echo " Error! Please enter the valid code or not same player from the
list and please
choose three players code only with space.For eg. PK VK DW"
            echo ""
            PlayerInfo  #PlayerInfo has been called
        fi
    }
    FileRead() {  #this function checks wheter the file exits or not.
        if [[ -f $1 ]]
        then
            cat $1
        else
            echo ""
            echo "Sorry !! File not found !!!"
            echo ""
            CountryList
        fi
    }
    Restart() {  #this function asks user whether he/she wants to continue or not
        echo ""
        echo -e "Do you continue the program. If continue enter the 'yes':\c"
        read answer
        if [ $answer == yes ]
        then
            echo ""
            echo " ⁂ Welcome back. Good Luck !!⁂"
            CountryList
        else
            echo "===[»»»» Thanks for playing !! »»»»»»»"
            echo "===[»»»» Have a good day.  »»»»»»» "
            exit
        fi
}
```

```
    Secretkey="torilal"
    if [[ $1 != ""  && $2 != "" ]]
    then
        while [[ $try -lt 5 ]]      #Here while loop is used to control flow input statement
            do
            echo -e "Enter the secretkey: \c"
            read password
            if [[ $password == $Secretkey ]]     #Here if is used to compare the secretkey provided by user with the assigned value
            then
            Theme $1 $2
            CountryList
            break
            else
            echo "Please try again. You have entered wrong secretkey. "
            let try++
            fi
            done
            echo ""
            echo "Please restart game again.Too many wrong attempts"
    else
```

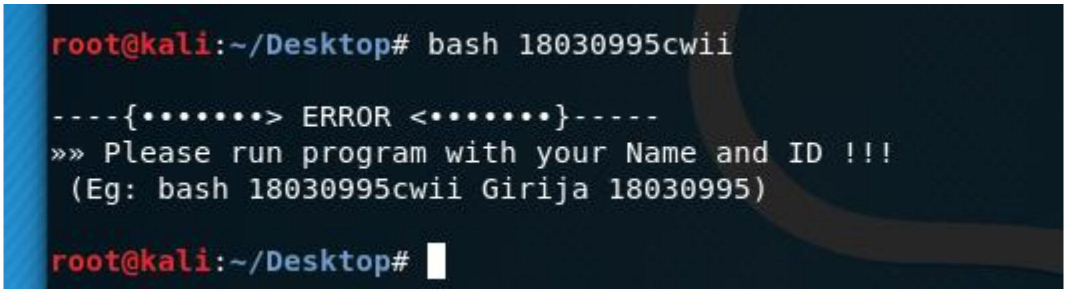
```
            echo " "

            echo "----{•••••••> ERROR <•••••••}-----"

            echo "»» Please run program with your Name and ID !!! "

            echo " (Eg: bash 18030995cwii Girija 18030995)"

            echo " "

    fi

    exit

}
```

3. Testing

Test 1: Running script file without username and id

Input:	bash 18030995cwii
Expected Result:	Error message will be displayed.
Actual Output:	Error message displayed as per expected result.
Test Result:	Test successful

Table 1: Running Script file without username and id



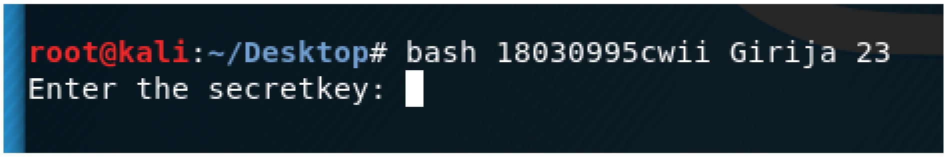
```
root@kali:~/Desktop# bash 18030995cwii

----{•••••••> ERROR <•••••••}-----
»» Please run program with your Name and ID !!!
 (Eg: bash 18030995cwii Girija 18030995)

root@kali:~/Desktop# 
```

Figure 1:Running Script file without username and id

Test 2. Running script file with username and id

Input:	bash 18030995cwii Girija 23
Expected Result:	Program shall ask the secret key to run the program
Actual Output:	As per expected result program ask secret key
Test Result:	Test successful

Table 2: Running Script file with username and id



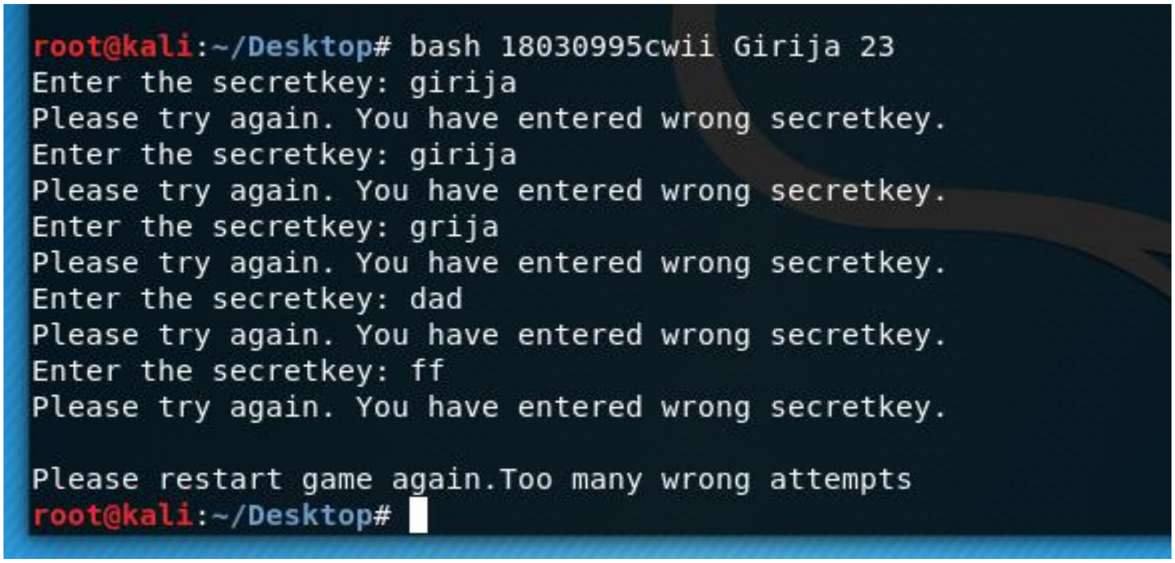
```
root@kali:~/Desktop# bash 18030995cwii Girija 23
Enter the secretkey: 
```

Figure 2: Running Script file with username and id

Test 3. Entering wrong password for 5 times

Input:	Entering wrong password like girija, dad etc
Expected Result:	Program shall provide five chances to enter right password
Actual Output:	As per expected result program ask password for five times and exits the program.
Test Result:	Test successful

Table 3: Entering wrong password for five times.



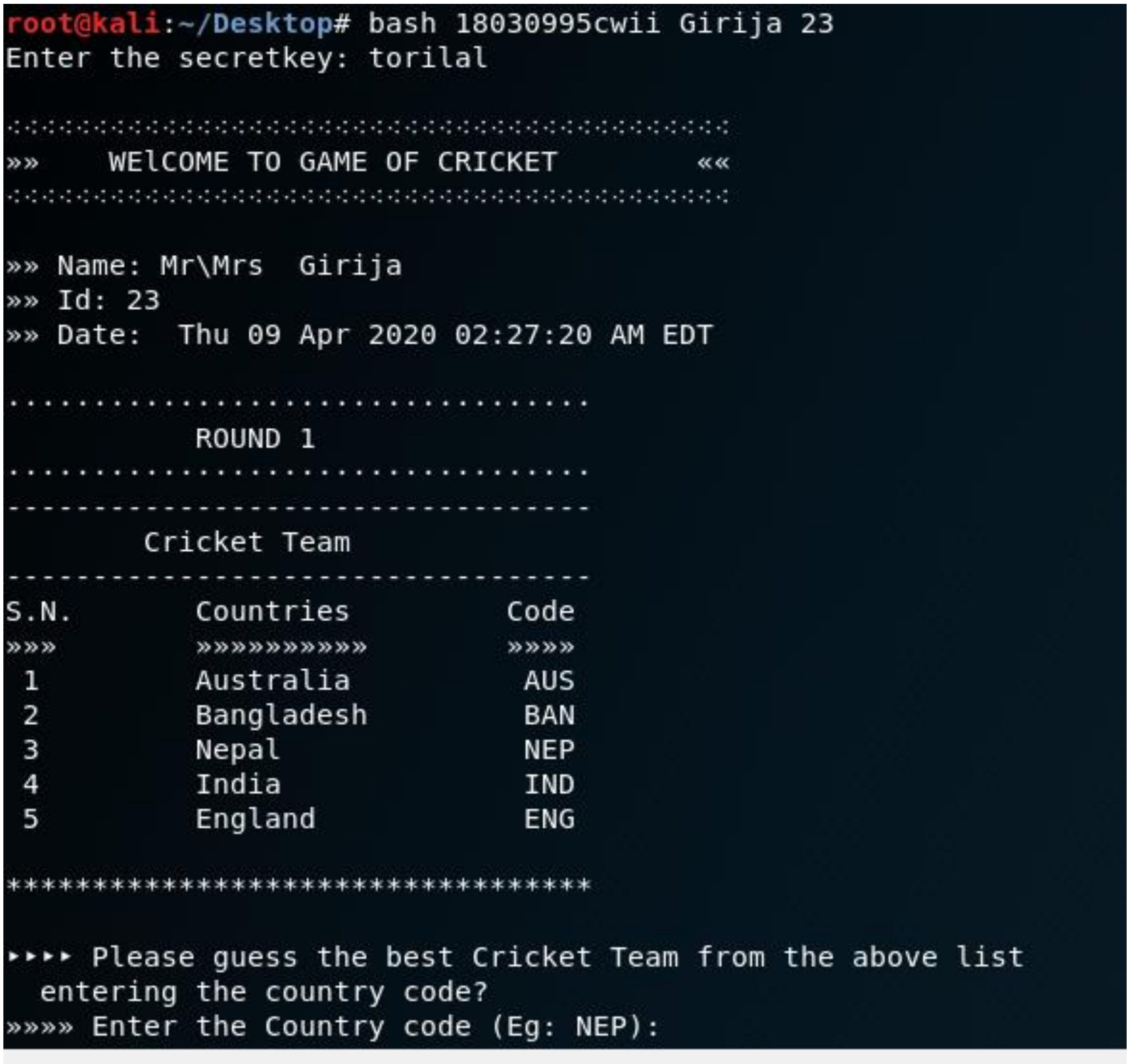
```
root@kali:~/Desktop# bash 18030995cwii Girija 23
Enter the secretkey: girija
Please try again. You have entered wrong secretkey.
Enter the secretkey: girija
Please try again. You have entered wrong secretkey.
Enter the secretkey: grija
Please try again. You have entered wrong secretkey.
Enter the secretkey: dad
Please try again. You have entered wrong secretkey.
Enter the secretkey: ff
Please try again. You have entered wrong secretkey.

Please restart game again.Too many wrong attempts
root@kali:~/Desktop# 
```

Figure 3: Entering wrong password for five times.

Test 4. Entering right password in secret key.

Input:	Entering right password in secret key.
Expected Result:	Program welcomes the user with his/her id, name and display the date
Actual Output:	As per expected result program welcomes the user.
Test Result:	Test successful

Table 4: Entering right password in secret key.



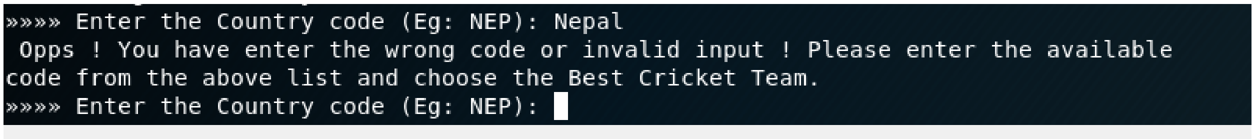
```
root@kali:~/Desktop# bash 18030995cwii Girija 23
Enter the secretkey: torilal

.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:
»»    WElCOME TO GAME OF CRICKET        ««
.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:.:

»» Name: Mr\Mrs  Girija
»» Id: 23
»» Date:  Thu 09 Apr 2020 02:27:20 AM EDT

..................................
           ROUND 1
..................................
----------------------------------
        Cricket Team
----------------------------------
S.N.       Countries         Code
»»»        »»»»»»»»»»        »»»»
 1         Australia          AUS
 2         Bangladesh         BAN
 3         Nepal              NEP
 4         India              IND
 5         England            ENG

**********************************

▸▸▸▸ Please guess the best Cricket Team from the above list
  entering the country code?
»»»» Enter the Country code (Eg: NEP):
```

Figure 4: Entering right password in secret key.

Test 5. Entering country name

Input:	Entering country name
Expected Result:	Program will display the error message.
Actual Output:	As per expected result program displayed error message.
Test Result:	Test successful

Table 5: Entering country name.



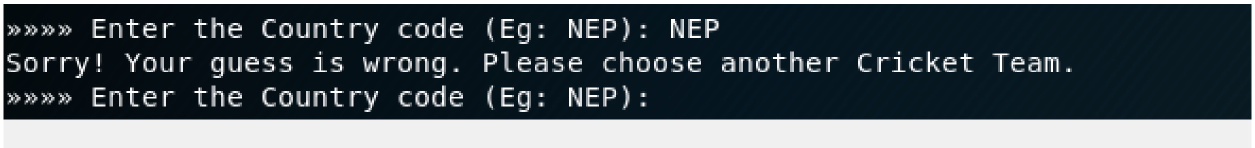
```
»»»» Enter the Country code (Eg: NEP): Nepal
 Opps ! You have enter the wrong code or invalid input ! Please enter the available
code from the above list and choose the Best Cricket Team.
»»»» Enter the Country code (Eg: NEP):
```

Figure 5:Entering country name.

Test 6. Giving incorrect country code

Input:	Entering incorrect country code.
Expected Result:	Program informs about wrong option.
Actual Output:	As per expected result program informs about wrong option.
Test Result:	Test successful

Table 6: Choosing incorrect country code.



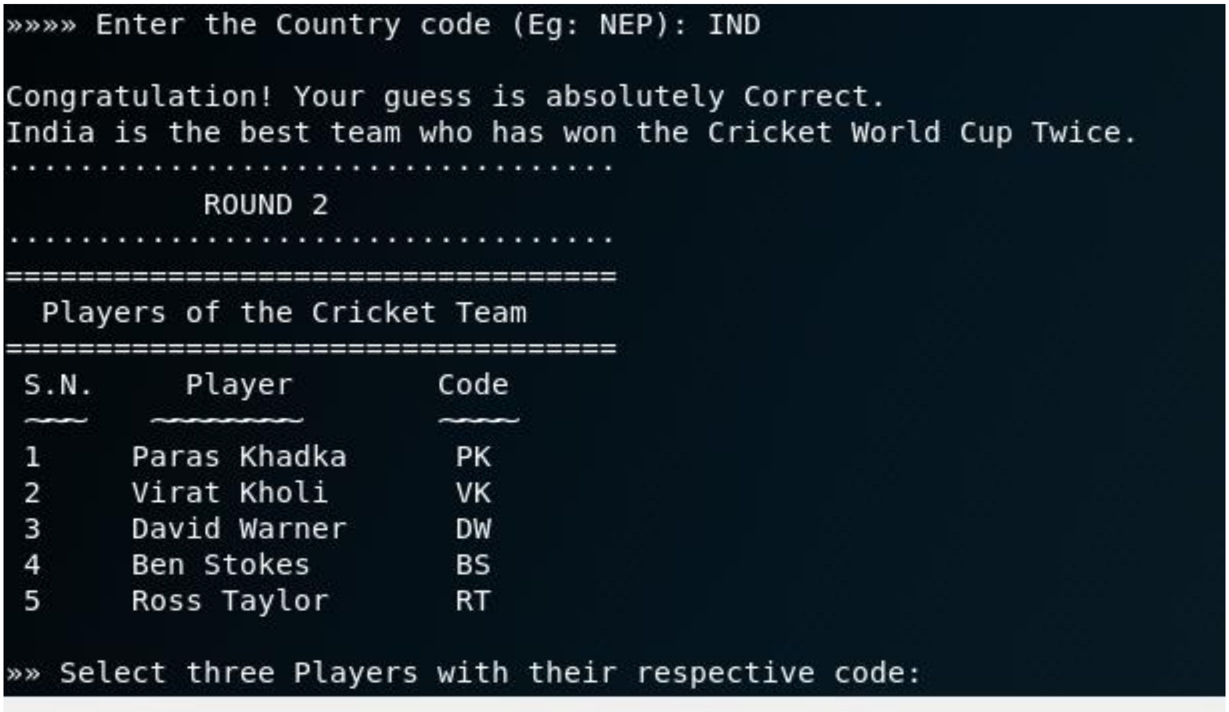
```
»»»» Enter the Country code (Eg: NEP): NEP
Sorry! Your guess is wrong. Please choose another Cricket Team.
»»»» Enter the Country code (Eg: NEP):
```

Figure 6:Choosing incorrect country code.

Test 7. Giving correct country code

Input:	Entering right country code.
Expected Result:	Program informs about choosing right option and display information about that team.
Actual Output:	As per expected result program display information about that team.
Test Result:	Test successful

Table 7: Giving correct country code.



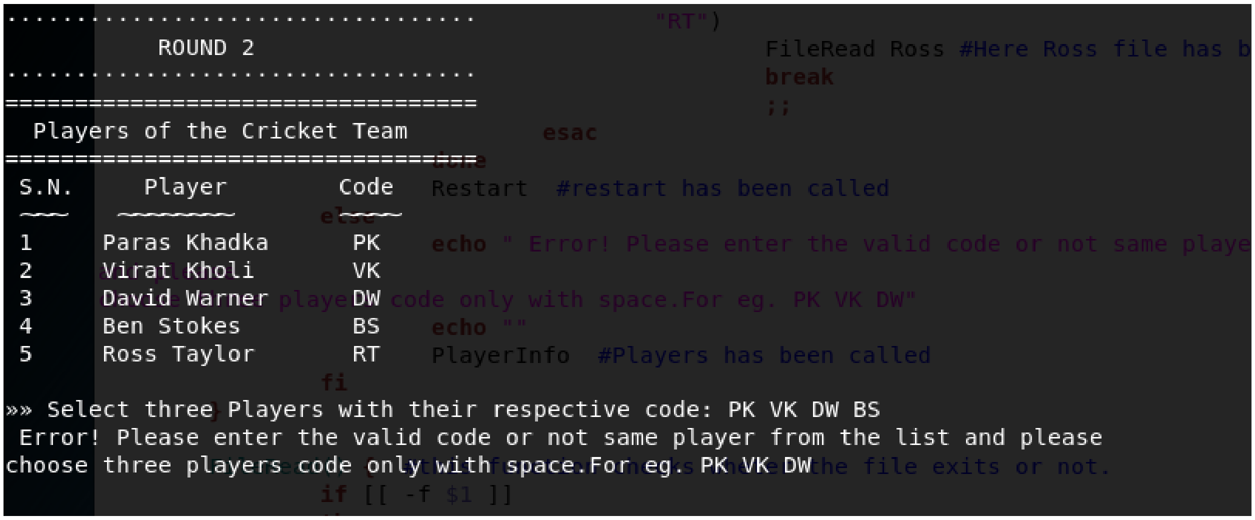
```
»»»» Enter the Country code (Eg: NEP): IND

Congratulation! Your guess is absolutely Correct.
India is the best team who has won the Cricket World Cup Twice.
..................................
           ROUND 2
..................................
==================================
  Players of the Cricket Team
==================================
 S.N.     Player        Code
 ~~~~   ~~~~~~~~~~      ~~~~~
 1     Paras Khadka      PK
 2     Virat Kholi       VK
 3     David Warner      DW
 4     Ben Stokes        BS
 5     Ross Taylor       RT

»» Select three Players with their respective code:
```

Figure 7: Giving correct country code.

Test 8. Selecting four players name

Input:	Entering four player code.
Expected Result:	Program informs about invalid input given.
Actual Output:	As per expected result program informs about the invalid input.
Test Result:	Test successful

Table 8: Selecting four players name.



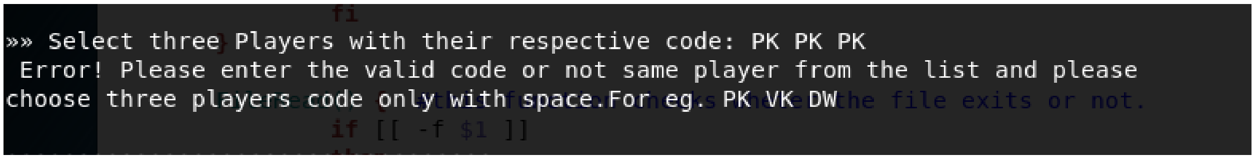
```
..................................
        ROUND 2
..................................
==================================
  Players of the Cricket Team
==================================
 S.N.     Player        Code
 ~~~    ~~~~~~~~~      ~~~~~
 1     Paras Khadka      PK
 2     Virat Kholi       VK
 3     David Warner      DW
 4     Ben Stokes        BS
 5     Ross Taylor       RT

»» Select three Players with their respective code: PK VK DW BS
 Error! Please enter the valid code or not same player from the list and please
choose three players code only with space.For eg. PK VK DW
```

Figure 8: Selecting four players name.

Test 9. Selecting same player code.

Input:	selecting same payer code
Expected Result:	Program informs user about invalid option given.
Actual Output:	As per expected result program informs about invalid option.
Test Result:	Test successful

Table 9: Selecting same player code.



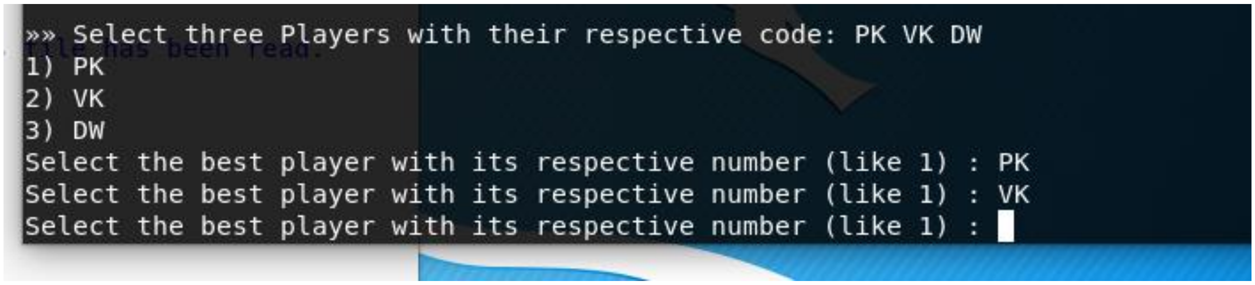
```
»» Select three Players with their respective code: PK PK PK
 Error! Please enter the valid code or not same player from the list and please
choose three players code only with space.For eg. PK VK DW
```

Figure 9: Selecting same player code.

Test 10. Selecting wrong player id

Input:	Entering wrong player id
Expected Result:	Program provides chance to enter right id again.
Actual Output:	As per expected result program provides another chance to the user.
Test Result:	Test successful

Table 10: selecting wrong player id.



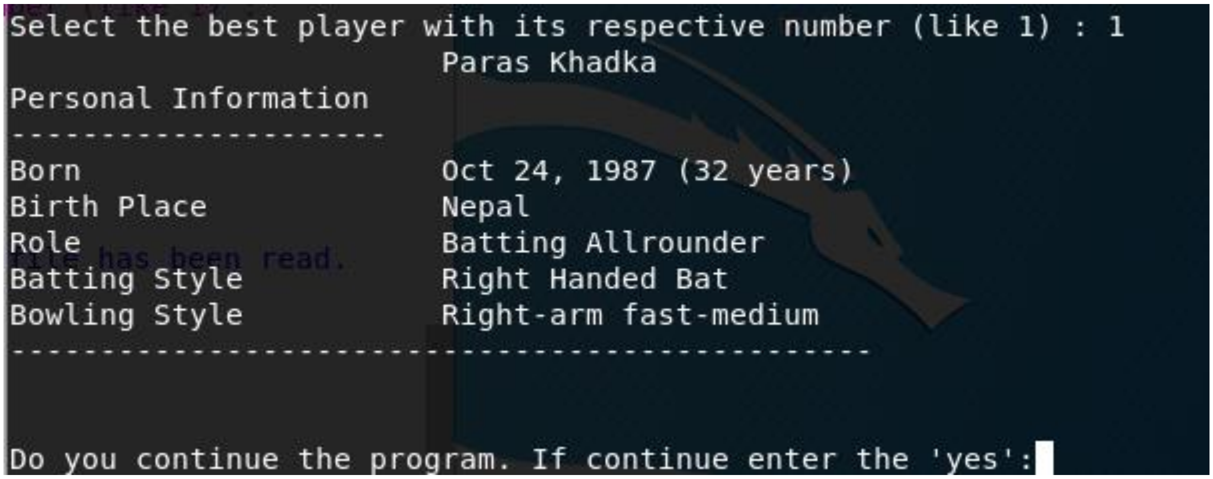
```
»» Select three Players with their respective code: PK VK DW
1) PK
2) VK
3) DW
Select the best player with its respective number (like 1) : PK
Select the best player with its respective number (like 1) : VK
Select the best player with its respective number (like 1) : 
```

Figure 10: selecting wrong player id.

Test 11. Selecting right player id

Input:	Selecting right player id
Expected Result:	Program provide information about selected player.
Actual Output:	As per expected result program display information about player.
Test Result:	Test successful

Table 11: Selecting right player id.



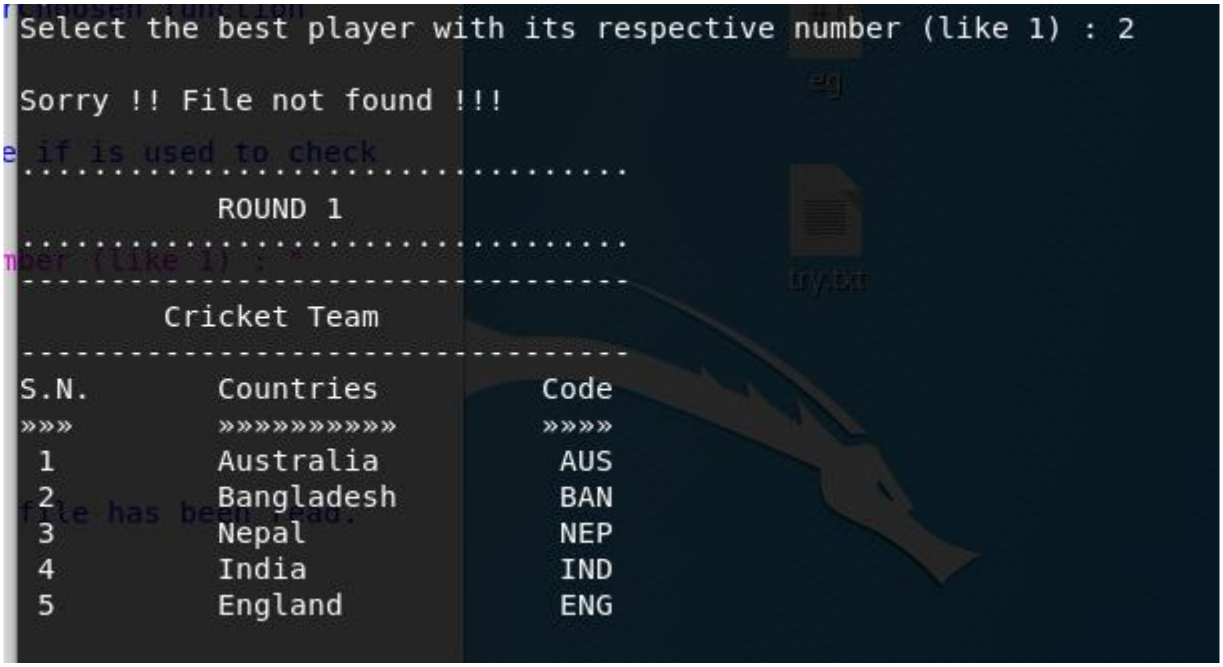
```
Select the best player with its respective number (like 1) : 1
                            Paras Khadka
Personal Information
---------------------
Born                        Oct 24, 1987 (32 years)
Birth Place                 Nepal
Role                        Batting Allrounder
Batting Style               Right Handed Bat
Bowling Style               Right-arm fast-medium
------------------------------------------------


Do you continue the program. If continue enter the 'yes':
```

Figure 11: Selecting right player id.

Test 12. Selecting player without having profile

Input:	Selecting player without having profile.
Expected Result:	Program provide information about selected player.
Actual Output:	As per expected result program display information about player.
Test Result:	Test successful

Table 12: Selecting player without having profile.



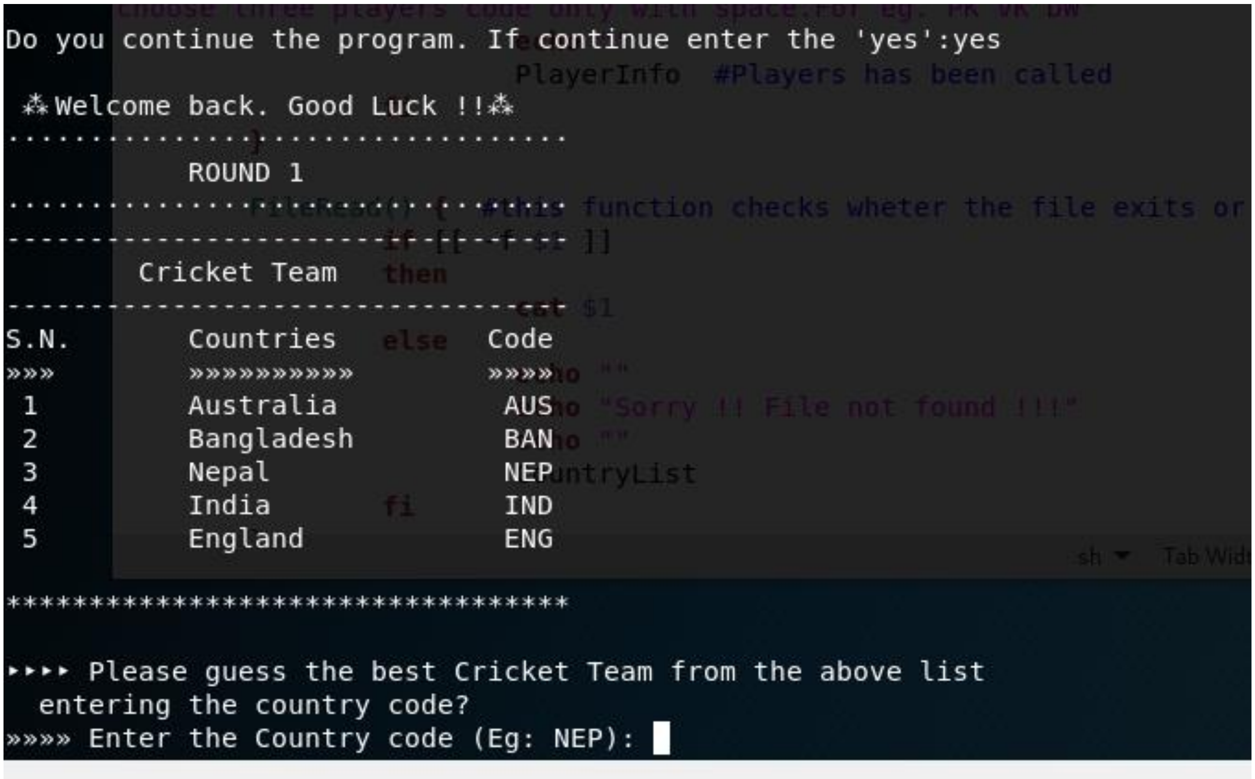
```
Select the best player with its respective number (like 1) : 2

Sorry !! File not found !!!
..................................
           ROUND 1
..................................
----------------------------------
        Cricket Team
----------------------------------
S.N.        Countries          Code
»»»         »»»»»»»»»»         »»»»
 1          Australia           AUS
 2          Bangladesh          BAN
 3          Nepal               NEP
 4          India               IND
 5          England             ENG
```

Figure 12: Selecting player without having profile.

Test 13. Continuing game

Input:	Selecting counting game option.
Expected Result:	Program restart again.
Actual Output:	As per expected result program restart again.
Test Result:	Test successful

Table 13: Continuing game.



```
Do you continue the program. If continue enter the 'yes':yes

 ⁂Welcome back. Good Luck !!⁂
..................................
           ROUND 1
..................................
----------------------------------
        Cricket Team
----------------------------------
S.N.        Countries         Code
»»»         »»»»»»»»»»        »»»»
 1          Australia          AUS
 2          Bangladesh         BAN
 3          Nepal              NEP
 4          India              IND
 5          England            ENG

**********************************

▸▸▸▸ Please guess the best Cricket Team from the above list
  entering the country code?
»»»» Enter the Country code (Eg: NEP): █
```

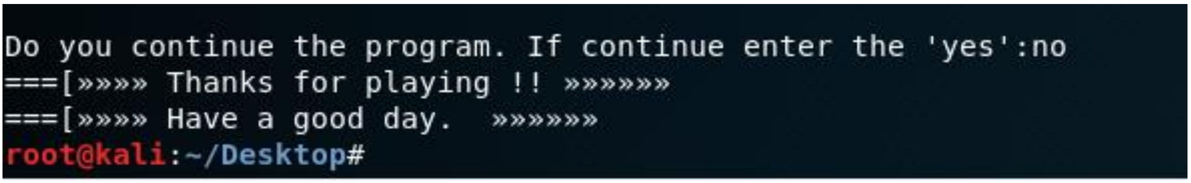
Figure 13: Continuing game.

Test 14. Exiting game

Input:	exiting game.
Expected Result:	Program provide information about exiting game.
Actual Output:	As per expected result program exits.
Test Result:	Test successful

Table 14: Exiting game.

:



```
Do you continue the program. If continue enter the 'yes':no
===[»»»» Thanks for playing !! »»»»»»»
===[»»»» Have a good day.  »»»»»»»
root@kali:~/Desktop#
```

Figure 14: Exiting game.

4. Content of three files

Player 1: Paras Khadka

Paras Khadka

Personal Information

Born	Oct 24, 1987 (32 years)
Birth Place	Nepal
Role	Batting Allrounder
Batting Style	Right Handed Bat
Bowling Style	Right-arm fast-medium

Player 2: Ben Strokes

Ben Stokes

Personal Information

Born	Jun 04, 1991 (28 years)
Birth Place	Canterbury, New Zealand
Height	--
Role	Batting Allrounder
Batting Style	Left Handed Bat
Bowling Style	Right-arm fast-medium

Player 3: David Warner

David Warner

Personal Information

Born	Oct 27, 1986 (33 years)
Birth Place	Paddington, New South Wales
Height	1.70 m
Role	Batsman
Batting Style	Left Handed Bat
Bowling Style	Right-arm legbreak

5. Conclusion

After numerous efforts and hardship, I was able to complete this coursework which was related about developing a user-friendly script or program in a Unix environment. As per the given scenario for doing task A, I have developed a program that implements interaction with the UNIX environment in a friendly manner and executes simple input/output operations. The tasks in the coursework is all about constructing an interactive program, checking user input errors, diagnosing those input error with clear messages and testing a program in Unix environment.

This coursework is based on UNIX Script Programming, so it helps me to gain basic knowledge on Unix shell environment and different Unix commands. It was hard to understand and tackle the scenario at first, but I was able to solve the scenario after some analysis and research. This coursework enhanced my personal skills and knowledge about Unix shell, script programming and learned various Unix commands. I got chance to do research work on Unix commands as well as understanding the shell operating system which helps me to enhance my knowledge and learn new things.

Task B

1. Introduction

This report is about “Memory Management” that focus on locality of reference in Memory Management. It also provides an overview the basic concepts of hierarchy of memory management, memory allocation, static storage allocation and dynamic storage allocation. It also describes briefly about of paging and segmentation.

Memory Management is the method of managing and organizing computer memory, allocating portions known as blocks to different running programs to maximize the overall system performance. It allows OS to keep track of any location of the memory, regardless of whether it is assigned to some process or remains free. It checks how much memory is to be allocated to processes (Krishna, 2018).

2. Aim and Objective

The main aims of preparing this report are as follows:

- To provide concept of Memory Management.
- To provide idea on locality of reference in Memory Management.
- To provide basic concept of hierarchy of memory management and memory allocation.
- Gain idea about static and dynamic storage allocation.
- To know briefly about of paging and segmentation.
- Know advantages and disadvantages of paging and segmentation.

The objectives of this report are as follows:

- To make familiar about memory management in Unix.
- Giving knowledge on locality of reference in Memory Management.
- To explore idea on hierarchy of memory management and memory allocation.
- Explain briefly about static and dynamic storage allocation.
- To look about different memory management schemes,
- Describe briefly about paging and segmentation with its advantages and disadvantages of paging and segmentation

3. Background

Memory is an important resource in computer. Memory management is the computer memory management process which consists of the primary memory and the secondary memory. The memory management aim is to keep track of which memory parts are in use and which parts are not in use, to assign memory to processes when they need it, and to assign it when it is finished. UNIX memory management scheme involves swapping and paging of demands (Liu & Yong, 2011).

Hierarchy of memory management

Memory hierarchy is a term used in computer architecture to address performance issues in computer architectural design, algorithm forecasts, and lower level programming constructs involving reference locality. Computer-storage memory hierarchy divides each of its tiers based on response time. The memory in a computer can be separated into five hierarchies based on both speed and use. The five memory hierarchies are registers, cache, main memory, magnetic disks, and magnetic tapes. The first three hierarchies are unreliable memories that indicate that there is no control and then they lose their stored data automatically. Whereas the last two hierarchies are not dynamic, meaning they permanently store the data (Elprocus, 2017). The figure given below demonstrates the different levels of memory hierarchy:

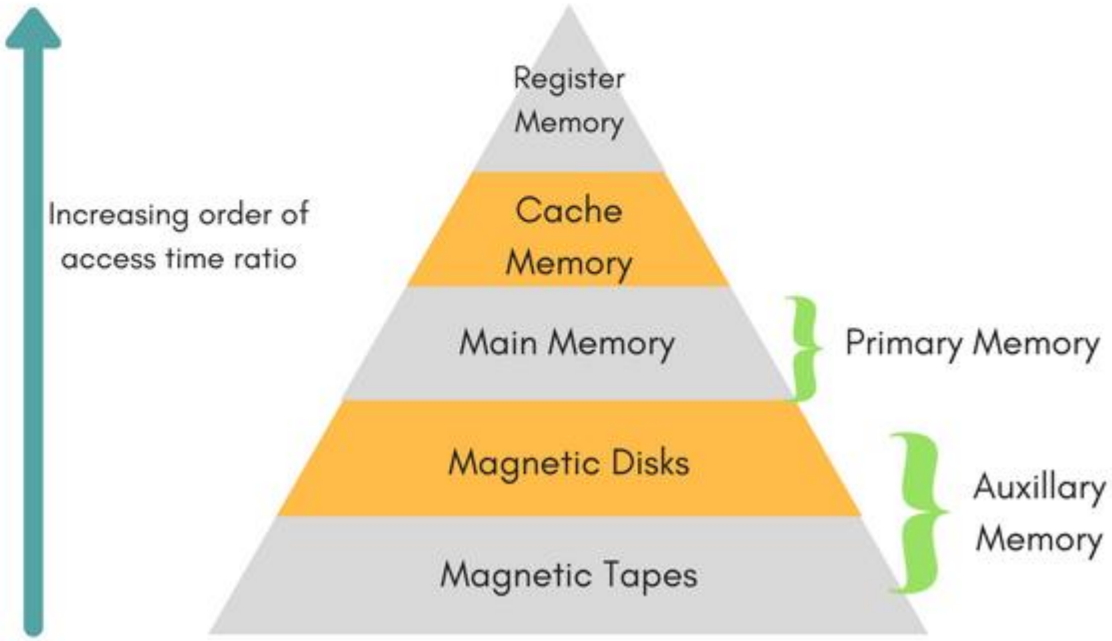


Figure 15: Hierarchy of memory management.

Locality of references refers to a phenomenon where a computer program appears to access the same set of memory locations over a given time interval. The locality property of reference is primarily seen in a program by loops and subroutine calls.

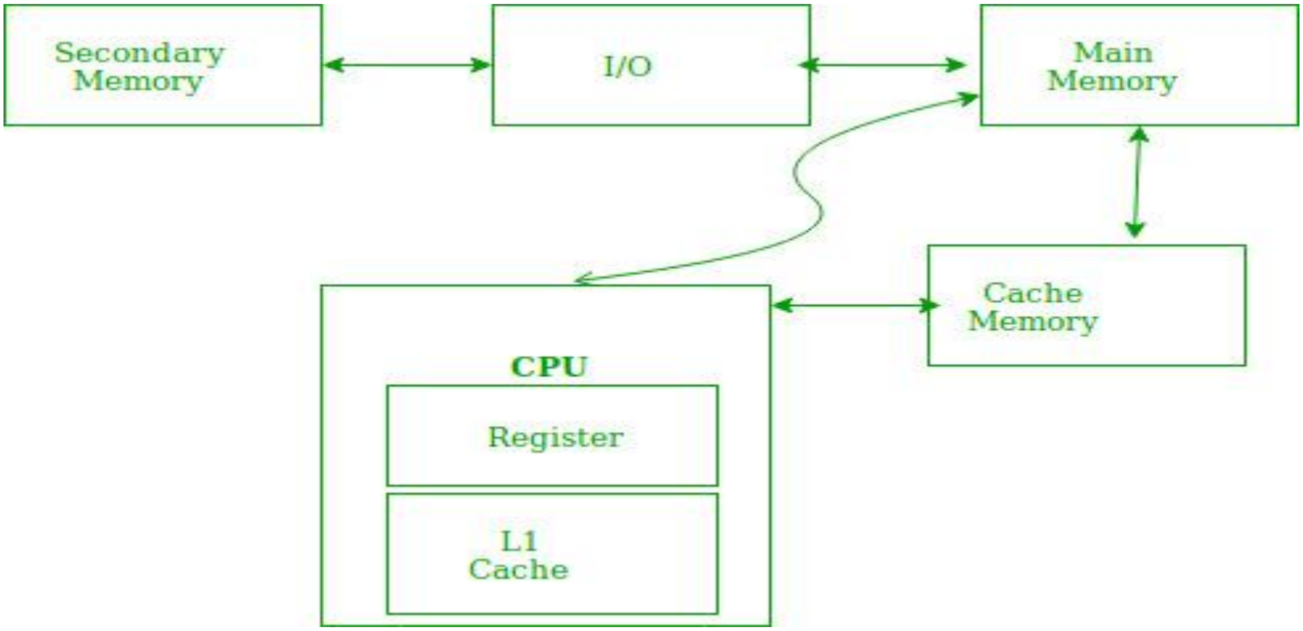


Figure 16: Locality of references

Cache Operation is based on the principle of locality of reference (geeksforgeeks, 2019). There are two ways to collect data or instructions from the main memory and store them in the memory of the cache. These two ways are the given below:

Temporal Locality

As Processor accesses the current main memory location for reading needed data or instructions, it will also be stored in the cache memory based on the assumption that in the near future the same data or instructions will be needed. This is known as temporal locality.

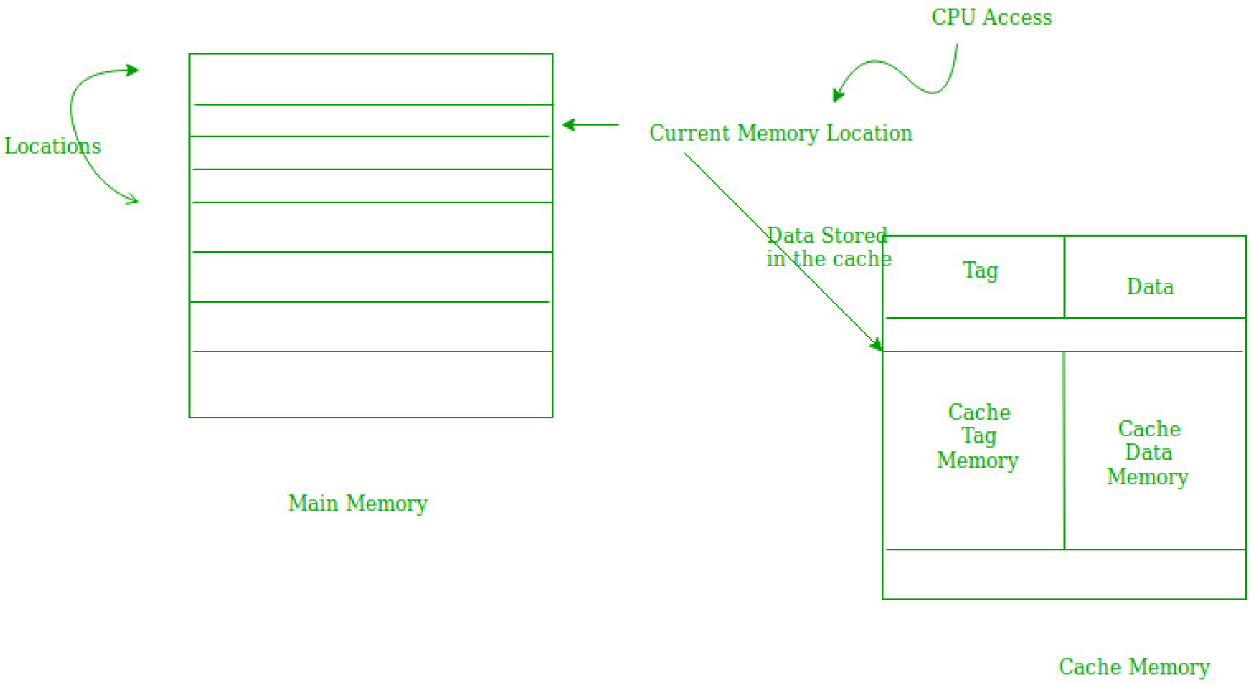


Figure 17:Temporal locality

Temporal locality means that it will be appropriate to obtain current data or instructions in the near future. So, we can store the data or instruction in the cache memory so we can stop looking for the same data again in main memory.

Spatial Locality:

If at a given time a specific storage location is referenced, then similar memory locations are likely to be referenced in the near future. In this case it is common to attempt to estimate the size and shape of the region around the current reference for which planning faster access for subsequent reference is worthwhile. This is slightly different from the temporal locality. Here we are talking about nearly located memory locations when we were talking about the actual location of the memory that was being stored in temporary location (Stallings, 2010).

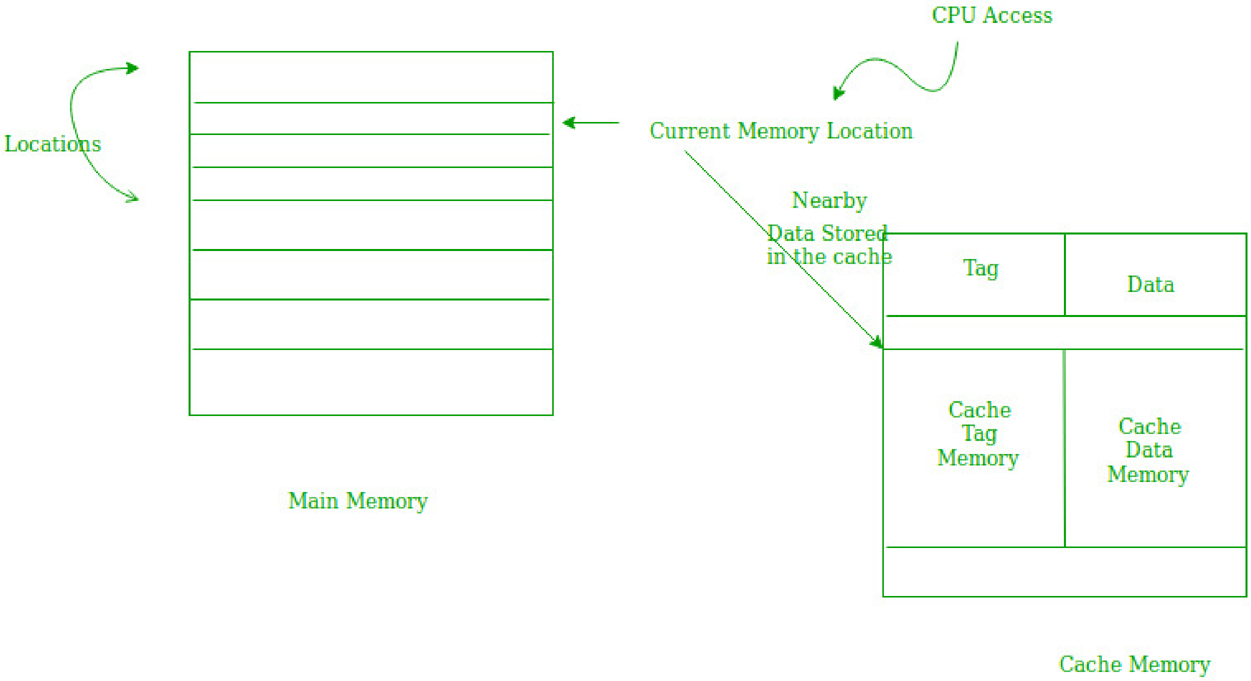


Figure 18: Spatial locality.

Memory Allocation

Memory allocation is the method of reserving a part or whole portion of computer memory for program and task execution. Memory allocation is accomplished by means of a mechanism called memory management. In physical and virtual memory management the mechanism of allocating memory is very common. Once executed, the programs and facilities are allocated with a particular memory according to their specifications. The memory is released and transferred to another program or combined within the primary memory until the program has ended its operation or is idle (Neha, 2019). There are two types of memory allocation. They are Static Memory Allocation and Dynamic Memory Allocation.

Static Memory Allocation

Static memory allocation is the allocation of memory at compile time, before the associated program is executed. Once the compiler compiles the program and produces object files, the static memory allocation is performed, the linker merges all these object files and creates a single executable file, and the loader loads this single executable file into main memory for execution. For static memory allocation, the size of the data needed for the process must be known before process execution begins. Static method of allocating memory does not require any operation of allocating memory during process execution. As all the memory allocation operation needed for the process is performed before process execution has begun. So, it leads to faster execution of a process (Manomaya, 2016).

Dynamic Storage Allocation

Dynamic memory allocation is performed while the program is in execution. Here the memory is allocated to the program entities as they are to be used for the first time during running of the program. The actual size of the necessary data is determined at runtime, and the exact memory space is allocated to the program, eliminating memory waste.

Dynamic memory allocation enables flexibility in program execution. As it can determine which amount of memory space the program will need. If the program is big enough then on the various parts of the program a dynamic memory allocation is performed and is to be used at present. This reduces memory wastage and increases system efficiency. Allocation of dynamic memory doesn't require special operating system support. It is the programmer's duty to construct the program in a way that takes advantage of the process of allocating dynamic memory. So, the allocation of dynamic memory is versatile, but slower than static allocation of memory (T, 2019).

To maximize resource allocation the operating system uses multiple memory management schemes. Those schemes are responsible for allocating memory chunks depending on the operating system's requirement. The operating system uses different schemes for this purpose, the most common among them are, first fit, best fit, and worst fit (Knuth, 1997).

First- fit

The first fit allocation scheme for memory inspects the empty memory block sequentially. This means that the memory block that was found empty is tested for size at the first attempt; if the size is not less than the size needed, it is allocated. One of the main problems with this memory allocation scheme is that it generates large chunks of memory space when a process is allocated to a comparatively bigger space than required (PrepInsta, 2019).

Best – fit

The operating system looks for the empty memory block in case of the best suited memory allocation scheme. Once the operating system identifies the memory block with least memory wastage, this is allocated to the operation. This scheme is considered the best approach, because it results in the most efficient allocation of memory. However, it can be time consuming to find the best match memory allocation.

Worst – fit

In the case of the worst fit memory allocation scheme, the operating system is looking for free memory blocks which the operating system needs. The processes are allocated an empty block until detected by the CPU. The scheme is often considered to be the worst suited memory management scheme because often a process is allocated a memory block that is much larger than the actual demand resulting in a massive amount of memory wasted (PrepInsta, 2019).

Description of paging and segmentation

Paging is a storage mechanism that enables OS to retrieve processes in the form of pages from the secondary storage into the main memory. The main memory in the Paging system is divided into small fixed-size physical memory blocks which are called frames. To have optimum use of the main memory and to prevent external fragmentation, the size of a frame should be kept the same as that of a file. To have optimum use of the main memory and to prevent external fragmentation, the size of a frame should be kept the same as that of a file. Paging is used to allow data access easier and is a logical term.

Advantages of Paging:

- Easy to use algorithm for Memory Management
- No need for external Fragmentation
- Swapping between pages of equal size and page frames is easy

Disadvantages of Paging

- May cause Internal fragmentation
- Complex algorithm for memory management
- Page tables consume additional memory.
- Multi-level paging may lead to memory reference overhead (Krishna, 2018).

Segmentation is a memory management technique in which, the memory is divided into the variable size parts. The process of segmentation operates much like paging, the only difference between the two being that the segments are of variable length while the pages are still of fixed size in the paging method. A portion of the program contains the key component of the program, data structures, utility functions etc. The OS maintains a map table for all processes inside the section. This also includes a list of free memory blocks along with their size, section numbers and memory locations in the main memory or virtual memory.

Advantages of Segmentation

- There is no internal division
- Average Segment Size is larger than the actual page size.
- Less overhead
- The segments can be moved more easily than the entire address space.
- The segment table is of lesser size as compare to the page table in paging.

Disadvantages of Segmentation

- It can have external fragmentation.
- Contiguous memory can hardly be allocated to a variable partition size.
- Costly memory management algorithms (JavaTpoint, 2018).

4. Conclusion

Memory Management is a broad term that incorporates all the process of intelligently managing all the operations and resources associated with resource in the primary memory or storage disk when there are several processes that use memory and resources. This keeps track of all the memory, it may be idle in the system, or at a given time it may be allocated to different processes.

In this report work I have tried to give and overview of the concept of memory management. I also tried to provide the basic concepts of hierarchy of memory management, memory allocation, static storage allocation and dynamic storage description. This report also describes briefly about of paging and segmentation with its advantages and disadvantages.

Nearly all a computer programmer does include that he or she understand how memory should be handled. Even storing a number in memory allows the programmer to decide how it should be handled by the computer.

At last, proper memory management is essential to every computer operating system. Modern advance operating system has appropriate guidelines for allocating their memory to various tasks during any program's execution. Failure to follow memory allocation rules will result in bugs, poor results and, in the worst case, viruses and malicious software taking over.

5. References

Ahlawat, A., 2019. *Introduction to Memory Management.* [Online] Available at: https://www.studytonight.com/operating-system/memory-management [Accessed 02 April 2020].

Elprocus, 2017. *What is Memory Hierarchy?.* [Online] Available at: https://www.elprocus.com/memory-hierarchy-in-computer-architecture/ [Accessed 02 April 2020].

geeksforgeeks.org, 2018. *Introduction to Linux Shell and Shell Scripting.* [Online] Available at: https://www.geeksforgeeks.org/introduction-linux-shell-shell-scripting/ [Accessed 01 April 2020].

geeksforgeeks, 2019. *Locality of Reference and Cache Operation in Cache Memory.* [Online]
Available at: https://www.geeksforgeeks.org/locality-of-reference-and-cache-operation-in-cache-memory/
[Accessed 02 April 2020].

Janssen, D. & Janssen, C., 2017. *Definition - What does Shell mean?.* [Online] Available at: https://www.techopedia.com/definition/3427/shell [Accessed 01 April 2020].

JavaTpoint, 2018. *Segmentation.* [Online] Available at: https://www.javatpoint.com/os-segmentation [Accessed 04 April 2020].

Knuth, D., 1997. *Fundamental Algorithms.* 5th ed. Boston: Addison-Wesley.

Krishna, 2018. *Paging in Operating System(OS).* [Online] Available at: https://www.guru99.com/paging-in-operating-system.html#1 [Accessed 03 April 2020].

Liu, Y. & Yong, Y., 2011. *UNIX Memory Management.* 2nd ed. Berlin: Springer Higher Education Press.

Manomaya, T., 2016. *What is static memory allocation and dynamic memory allocation?.* [Online]
Available at: https://www.manomayasoft.com/blog/item/134-what-is-static-memory-allocation-and-dynamic-memory-allocation
[Accessed 03 April 2020].

Neha, T., 2019. *Memory Allocation.* [Online]
Available at: https://binaryterms.com/static-and-dynamic-memory-allocation.html#MemoryAllocation
[Accessed 03 April 2020].

PrepInsta, 2019. *First Fit in OS (Operating System).* [Online]
Available at: https://prepinsta.com/operating-systems/page-replacement-algorithms/first-fit/
[Accessed 04 April 2020].

PrepInsta, 2019. *Worst Fit Algorithm in Operating System (OS).* [Online]
Available at: https://prepinsta.com/operating-systems/page-replacement-algorithms/worst-fit/
[Accessed 04 April 2020].

PrepInsta, 2019. *Worst Fit in OS (Operating System).* [Online]
Available at: https://prepinsta.com/operating-systems/page-replacement-algorithms/worst-fit/
[Accessed 04 April 2020].

Stallings, W., 2010. *Computer organization and architecture : designing for performance.* s.l.:s.n.

Swati, 2019. *What Is Unix: A Brief Introduction To Unix.* [Online]
Available at: https://www.softwaretestinghelp.com/unix-introduction/
[Accessed 19 March 2020].

T, N., 2019. *Types of Memory Allocation.* [Online]
Available at: https://binaryterms.com/static-and-dynamic-memory-

allocation.html#TypesofMemoryAllocation
[Accessed 03 April 2020].