



Module Code & Module Title

CC4057NT Introduction to Information System

Assessment Weightage & Type

20% Individual Coursework

Year and Semester

2019-20 Autumn

Student Name: sujan chaudhary

London Met ID:19031885

College ID: np05cp4a190172@iic.edu.np

Assignment Due Date:2021/04/11

Assignment Submission Date:2021/04/10

Title (Where Required):

Word Count (Where Required):

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 content

Contents

Task A.....	1
1.0. Introduction.....	1
2.0. script	2
3.0. Testing.....	13
3.0.1. Test 1 run without name	13
3.0.2. Test 2 run with more than 2 arguments	13
3.0.3. Test 3 run with username and id	14
3.0.5. Test 5 run correct password	15
3.0.6. test 6 country name	16
3.0.7. test 7 incorrect country CODE	17
3.0.8. test 8 correct country CODE but the wrong selection	18
3.0.9. test 9 correct country CODE	19
3.0.10. test 10 pick 4 player names	20
3.0.11. Test 11 pick same player name.....	21
3.0.12. TEST 12 DIFFERENT PLAYER NAME	22
3.0.13. Test 13 wrong user id.....	23
3.0.14. test 14 PROFILE NA BHAKO PLAYER ID.....	24
3.0.15. Test 15 right user-id	25
3.0.16. Test 16 EXIT YES	26
3.0.17. Test 17 EXIT NO.....	26
4.0. Contents of three files: (TEXTS) 6	27
• LM	27
• NJ	27
• KC.....	28
5.0. Conclusion	28
Task B.....	29
6.0. Introduction.....	29
7.0. Aims and objectives	30
8.0. Background.....	30
8.1.1. Physical memory	30

8.1.2.	Memory placement.....	31
8.1.3.	Page coloring.....	33
8.1.4.	Paging	34
8.1.5.	Segmentation.....	35
8.1.6.	Page size variation.....	37
9.0.	Conclusion	39
10.0.	References	40
11.0.	Appendix.....	41

Table of table

Table 1: Test 1 run without name.....	13
Table 2:Test 2 run with more than 2 arguments	13
Table 3:Test 3 run with username and id	14
Table 4:Test 4 run incorrect password 5 times	14
Table 5:Test 5 run correct password	15
Table 6:Test 6 country name	16
Table 7:Test 7 incorrect country CODE.....	17
Table 8:Test 8 correct country CODE but wrong selection.....	18
Table 9:Test 9 correct country CODE	19
Table 10:Test 10 pick 4 player names.....	21
Table 11:Test 11 pick same player name.....	22
Table 12:Test 12 DIFFERENT PLAYER NAME	23
Table 13:Test 13 wrong user id.....	24
Table 14:Test 14 PROFILE NA BHAKO PLAYER ID.....	24
Table 15:Test 15 right user id.....	25
Table 16:Test 16 EXIT YES.....	26
Table 17:Test 17 EXIT NO.....	27

Table of figure

Figure 1:Test 1 run without name.....	13
Figure 2:Test 2 run with more than 2 arguments.....	14
Figure 3:Test 3 run with username and id.....	14
Figure 4:Test 4 run incorrect password 5 times.....	15
Figure 5:Test 5 run correct password	16
Figure 6:Test 6 country name.....	17
Figure 7:Test 7 incorrect country CODE	18
Figure 8:Test 8 correct country CODE but wrong selection	19
Figure 9:Test 9 correct country CODE	20
Figure 10:test 10 pick 4 player names.....	21
Figure 11:test 11 pick same player name.....	22
Figure 12:test 12 DIFFERENT PLAYER NAME	23
Figure 13:Test 13 wrong user id.....	24
Figure 14:Test 14 PROFILE NA BHAKO PLAYER ID.	25
Figure 15:Test 15 right user id.....	26
Figure 16:Test 16 EXIT YES	26
Figure 17:Test 17 EXIT NO	27
Figure 18: first fit	32
Figure 19:best fit	32
Figure 20:worst fit	33
Figure 21:paging.....	35
Figure 22:segmentation	37

Task A**1.0. Introduction**

Bash stands for Bourne Again Shell is a type of interpreter that processes shell commands. A shell interpreter takes commands in plain text format and calls Operating System services to execute. For example, the ls command lists the files and folders in a directory, cd command changes the directory. Bash is the upgraded version of sh (Bourne Shell). A shell scripting is writing a program for the shell to execute and a shell script is a file or program that the shell will execute.

Shell Scripting is a free and open-source computer program that runs on the Unix/Linux shell. Shell Scripting is a program that allows you to write a series of commands to be executed by the shell. It can condense long and repetitive command sequences into a single, simple script that can be saved and executed at any time, reducing programming time. This Shell Scripting tutorial will teach you all from the fundamentals of the Linux/Unix shell scripting software to advanced Shell Scripting concepts.

The different capabilities of the shell script are a shell script can automate the execution of many commands that would otherwise be entered manually in a command-line interface. This can be accomplished without the user having to initiate each command individually. Many features found only in complex programming languages, such as arrays, variables, and comments, are available in modern shell scripts. Many complex applications can be written using this method. Many complicated applications can be written in shell scripts using these features. But there is a problem i.e. shell script languages don't support classes, threading, etc. It is much more flexible to use loops, variables, etc for multiple tasks in a shell script. An example of this is a Unix shell script known as bash, which converts jpg images to png images. There is a shortcut provided by a shell script for a system command where command options, environment settings, or post-processing apply. This still allows the shortcut script to act as a Unix command.

Example;

```
#!/bin/bash
```

```
echo "hello world"
```

2.0. script

```
#!/bin/bash
```

```
# Required Functions
```

```
country_code() { #this function is for country codes
```

```
    echo -e "Please choose the best football team \n from following country?"
```

```
    echo "-----"
```

```
    echo "| Code      | Country      |"
```

```
    echo "-----"
```

```
    echo "| BRZ        | Brazil       |"
```

```
    echo "-----"
```

```
    echo "| ARG        | Argentina    |"
```

```
    echo "-----"
```

```
    echo "| NEP        | Nepal        |"
```

```
    echo "-----"
```

```
    echo "| CHI        | China        |"
```

```
    echo "-----"
```

```
    echo "| ENG        | England      |"
```

```
    echo "-----"
}

players_code(){ #this function is for player codes

    echo "The list of best player of dear club of tarahara: " #use echo command

    echo "-----"

    echo "| Code   | Player       |"

    echo "-----"

    echo "| LM     | Lionel Messi |"

    echo "-----"

    echo "| NJ     | Neymar Junior |"

    echo "-----"

    echo "| KC     | Kiran Chemjong |"

    echo "-----"

    echo "| ZZ     | Zheng Zhi     |"

    echo "-----"

    echo "| HK     | Harry Kane     |"

    echo "-----"

    player_choose

}
```



```
player_choose(){ #this is function for choosing the player
```

```
    echo "#####"
```

```
    echo "#      PLEASE!! CHOOSE THREE PLAYER      #" 
```

```
    echo "#####"
```

```
    read a b c
```

```
    if [[ $a = $b || $a = $c || $b = $a || $b = $c || $c = $a || $c = $b  ]]
```

```
    then
```

```
        echo ""
```

```
        echo "#####"
```

```
        echo "#      ERROR!! PLAYER CODE IS BEING REPEATED      #" 
```

```
        echo "#####"
```

```
        player_choose
```

```
    else
```

```
        for player in $a $b $c
```

```
        do
```

```
            if ! [[ $player == LM || $player == NJ || $player == KC || $player ==  
ZZ || $player == HK ]]
```

```
            then
```

```
                echo
```

```
                "#####"
```

```
        echo "#      ERROR!! GIVEN CODE IS INVALID      #"

        echo

"#####"

        player_choose

        fi

    done

    num $a $b $c

fi

}

FileRead() {

    if [[ -f $1 ]]

    then

        cat $1

    else

        echo "#####"

        echo "#      ERROR!! THE FILE IS NOT AVAILABLE      #"

        echo "#####"

        players_code

    fi
```

```
}
```

```
num() {
```

```
    if [[ $# != 3 ]]; then
```

```
        echo "#####"
```

```
        echo "#      ERROR!! PLEASE GIVE ONLY THREE CODES      #"
```

```
        echo "#####"
```

```
        player_choose
```

```
    else
```

```
        PS3="PLEASE CHOOSE THREE PLAYERS AMONG FOLLOWING BY  
TYPING NUMBER:"
```

```
        select player in $a $b $c
```

```
        do
```

```
            case $player in
```

```
                "LM") FileRead LM #this will read file named LM
```

```
                break
```

```
                ;;
```

```
                "NJ") FileRead NJ #this will read file named NJ
```

```
                break
```

```
;;

"KC") FileRead KC #this will read file named KC

break

;;

"ZZ") FileRead ZZ #this will read file named ZZ

break

;;

"HK") FileRead HK #this will read file named HK

break

;;

*)                                                    echo
"#####"

        echo "#      ERROR!! PLEASE CHOOSE AMONG 1, 2, 3
NUMBERS ONLY    #"

        echo
"#####"

esac

done

echo ""

echo ""
```



```
        exit

    fi

fi

}

country_select(){ #this function is to select country

    country_code

    best_country=BRZ

    read country

    until [[ $country = $best_country ]]; do

        if [[ $country = ARG || $country = NEP || $country = CHI || $country = ENG
]]; then

            echo "$country is not the best football team."

            read country

        else

            echo

            "#####"

            echo "#    ERROR!! YOU HAVE TYPED INVALID CODE    #"

            echo

            "#####"

            read country
```

```
        fi

done

echo "Brazil is the country which have won the world cup for five times"

players_code

}

while [[ True ]]; do #passing two arguments

    if [[ $# -le 1 ]]; then

        echo "#####"

        echo "#      ERROR!! Please give both Name and ID      #"

        echo "#####"

        exit

    elif [[ $# -gt 2 ]]; then

        echo "#####"

        echo "#      ERROR!! Please give Name and ID only      #"

        echo "#####"

        exit

    fi

    secret_key=12345 #secret key for entering in program

    echo "#####"
```

```
echo "#   Please give the secret key for entrance   #"

echo "#####"

read -s key

count=4

while [[ $key != $secret_key ]]; do

    if [[ $count -ge 1 ]]; then

        echo

        "#####"

        echo "#   ERROR!! YOU HAVE ONLY $count CHANCE LEFT   #"

        echo

        "#####"

        ((count--)) #use of -- to decrease chance of giving secret key by 1

        read -s key #-s is used to hide secret key characters

    else

        echo

        echo

        "#####
        #####"

        echo "#   ERROR!! YOU HEVE ENTER WRONG SECRET KEY
        MORE THAN FOUR TIMES   #"

        echo

        "#####
        #####"
```



```

exit

fi

done

echo "##      ## ##### ##      ##### ##### ##      ## #####"

echo "## ## ## ##      ##      ##      ##      ## ##      ##      "

echo "## ## ## ##      ##      ##      ##      ## ##### ##### ##      "

echo "## ## ## ##### ##      ##      ##      ## ## ## ## ## ##### "

echo "## ## ## ##      ##      ##      ##      ## ##      ## ##      "

echo "## ## ## ##      ##      ##      ##      ## ##      ## ##      "

echo "### ### ##### ##### #####      #####      ##      ## #####"

echo "_____+++++++Hello! YOUR ID IS $2 AND YOUR NAME IS
$1+++++++_"

now="$(date)" #this show current time and date

printf "current time and date is %s--->\n" "$now"

echo ""

country_select

done

echo "-----"

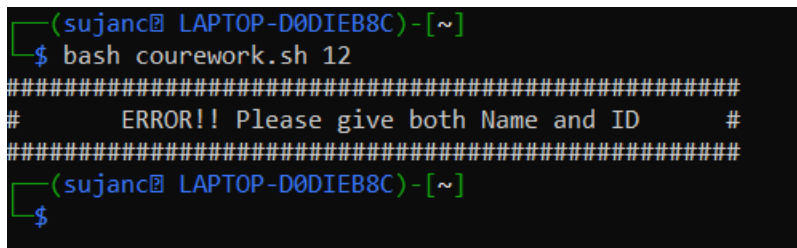
```

3.0. Testing

3.0.1. Test 1 run without name

Test no.	1
Input	Run the program without a name
Expected output	The error message should show and ask user to give both name and id
Actual	Error message showed and asked for name and id both
Test Result	Test successful

Table 1: Test 1 run without name



```

(sujanc@LAPTOP-D0DIEB8C)~$
$ bash courework.sh 12
#####
#      ERROR!! Please give both Name and ID      #
#####
(sujanc@LAPTOP-D0DIEB8C)~$
$

```

Figure 1: Test 1 run without name

3.0.2. Test 2 run with more than 2 arguments

Test no.	2
Input	run with more than 2 arguments
Expected output	The error message should show and the program should terminate itself
Actual	The error message showed and the program was terminated
Test Result	Test successful

Table 2: Test 2 run with more than 2 arguments

```

(sujanc@ LAPTOP-D0DIEB8C)-[~]
$ bash courework.sh sujan 12 sc
#####
#      ERROR!! Please give Name and ID only      #
#####
(sujanc@ LAPTOP-D0DIEB8C)-[~]
$

```

Figure 2: Test 2 run with more than 2 arguments

3.0.3. Test 3 run with username and id

Test no.	3
Input	Run the program without a name
Expected output	Ask for a secret key
Actual	Secret key was asked
Test Result	Test successful

Table 3: Test 3 run with username and id

```

(sujanc@ LAPTOP-D0DIEB8C)-[~]
$ bash courework.sh sujan 12
#####
#      Please give the secret key for entrance      #
#####

```

Figure 3: Test 3 run with username and id

3.0.4 Test 4 run incorrect password 5 times

Test no.	4
Input	Run incorrect password for 5 times
Expected output	The error message should show for 5 times
Actual	The error message showed for 5 times
Test Result	Test successful

Table 4: Test 4 run incorrect password 5 times

```
#####
#   Please give the secret key for entrance   #
#####
#   ERROR!! YOU HAVE ONLY 4 CHANCE LEFT   #
#####
#   ERROR!! YOU HAVE ONLY 3 CHANCE LEFT   #
#####
#   ERROR!! YOU HAVE ONLY 2 CHANCE LEFT   #
#####
#   ERROR!! YOU HAVE ONLY 1 CHANCE LEFT   #
#####
#   ERROR!! YOU HEVE ENTER WRONG SECRET KEY MORE THAN FOUR TIMES   #
#####
└─(sujanc@ LAPTOP-D0DIEB8C) - [~]
```

Figure 4: Test 4 run incorrect password 5 times

3.0.5. Test 5 run correct password

Test no.	5
Input	Run giving the correct password
Expected output	The program should run and ask to choose a country
Actual	The program was run and choose country was asked
Test Result	Test 0successful

Table 5: Test 5 run correct password

```
#####
#   Please give the secret key for entrance   #
#####
##   ## ##### ##   #####   #####   ##   ## #####
## ## ## ##   ##   ##   ##   ##   ##   ##
## ## ## ##   ##   ##   ##   ##   ##   ##   ##
## ## ## ##### ##   ##   ##   ##   ##   ##   ##
## ## ## ##   ##   ##   ##   ##   ##   ##   ##
## ## ## ##   ##   ##   ##   ##   ##   ##   ##
### ## ##### ##### #####   #####   ##   ## #####
_____+++++++Hello! YOUR ID IS 12 AND YOUR NAME IS sujan+++++++_____
current time and date is --->
Fri 09 Apr 2021 09:52:54 PM +0545
Please choose the best football team
from following country?
-----
| Code      | Country    |
-----
| BRZ       | Brazil     |
-----
| ARG       | Argentina  |
-----
| NEP       | Nepal      |
-----
| CHI       | China      |
-----
| ENG       | England    |
-----
```

Figure 5: Test 5 run correct password

3.0.6. test 6 country name

Test no.	6
Input	Run giving country name
Expected output	The error message should show and give another chance to give a valid country code
Actual	The error message showed and another chance was given to enter a valid code.
Test Result	Test successful

Table 6: Test 6 country name

```

Please choose the best football team
from following country?
-----
| Code      | Country    |
-----
| BRZ       | Brazil     |
-----
| ARG       | Argentina  |
-----
| NEP       | Nepal      |
-----
| CHI       | China      |
-----
| ENG       | England    |
-----
Brazil
#####
#          ERROR!! YOU HAVE TYPED INVALID CODE          #
#####

```

Figure 6: Test 6 country name

3.0.7. test 7 incorrect country CODE

Test no.	7
Input	Run the program giving incorrect country code
Expected output	The error message should show and give another chance to enter the correct country code
Actual	The error message showed and another chance was given to enter the correct country code
Test Result	Test successful

Table 7: Test 7 incorrect country CODE

```

Please choose the best football team
from following country?
-----
| Code      | Country    |
-----
| BRZ       | Brazil     |
-----
| ARG       | Argentina  |
-----
| NEP       | Nepal      |
-----
| CHI       | China      |
-----
| ENG       | England    |
-----
BRZA
#####
#          ERROR!! YOU HAVE TYPED INVALID CODE          #
#####

```

Figure 7: Test 7 incorrect country CODE

3.0.8. test 8 correct country CODE but the wrong selection

Test no.	8
Input	Run the program giving correct country code but the wrong selection
Expected output	The message should show and give another chance was given for correct selection of country code
Actual	The error message showed and another chance was given for the correct selection of country code
Test Result	Test successful

Table 8: Test 8 correct country CODE but wrong selection

```

Please choose the best football team
from following country?
-----
| Code      | Country    |
-----
| BRZ      | Brazil     |
-----
| ARG      | Argentina  |
-----
| NEP      | Nepal      |
-----
| CHI      | China      |
-----
| ENG      | England    |
-----
ARG
ARG is not the best football team.

```

Figure 8: Test 8 correct country CODE but wrong selection

3.0.9. test 9 correct country CODE

Test no.	9
Input	Run the program giving the correct country code
Expected output	The program should run and show a list of player code and ask to choose three players
Actual	The program was run and showed the list of player code and asked to choose three players
Test Result	Test successful

Table 9: Test 9 correct country CODE


```

Please choose the best football team
from following country?
-----
| Code      | Country    |
-----
| BRZ       | Brazil     |
-----
| ARG       | Argentina  |
-----
| NEP       | Nepal      |
-----
| CHI       | China      |
-----
| ENG       | England    |
-----
BRZ
Brazil is the country which have won the world cup for five times
The list of best player of dear club of tarahara:
-----
| Code      | Player      |
-----
| LM        | Lionel Messi |
-----
| NJ        | Neymar Junior |
-----
| KC        | Kiran Chemjong |
-----
| ZZ        | Zheng Zhi   |
-----
| HK        | Harry Kane   |
-----
#####
#          PLEASE!! CHOOSE THREE PLAYER          #
#####

```

Figure 9: Test 9 correct country CODE

3.0.10. test 10 pick 4 player names

Test no.	10
Input	Run the program by choosing 4 player names
Expected output	The error message should show to give only three codes and ask a user to choose three players
Actual	The error message showed to give only three codes and asked the user to choose three players.

Test Result	Test successful
-------------	-----------------

Table 10: Test 10 pick 4 player names

```

The list of best player of dear club of tarahara:
-----
| Code   | Player          |
-----
| LM     | Lionel Messi    |
-----
| NJ     | Neymar Junior   |
-----
| KC     | Kiran Chemjong  |
-----
| ZZ     | Zheng Zhi       |
-----
| HK     | Harry Kane      |
-----

#####
#      PLEASE!! CHOOSE THREE PLAYER      #
#####
LM NJ KC ZZ
#####
#      ERROR!! PLEASE GIVE ONLY THREE CODES      #
#####
#      PLEASE!! CHOOSE THREE PLAYER      #
#####

```

Figure 10: test 10 pick 4 player names

3.0.11. Test 11 pick same player name

Test no.	11
Input	Run the program by choosing 1 same player names
Expected output	Player code is repeated error message should show and ask a user to choose three players
Actual	Player code is repeated error message should show and ask a user to choose three players

Test Result	Test successful
-------------	-----------------

Table 11:Test 11 pick same player name

```

#####
# PLEASE!! CHOOSE THREE PLAYER #
#####
LM LM LM

#####
# ERROR!! PLAYER CODE IS BEING REPEATED #
#####
# PLEASE!! CHOOSE THREE PLAYER #
#####

```

The list of best player of dear club of tarahara:

Code	Player
LM	Lionel Messi
NJ	Neymar Junior
KC	Kiran Chemjong
ZZ	Zheng Zhi
HK	Harry Kane

Figure 11:test 11 pick same player name

3.0.12. TEST 12 DIFFERENT PLAYER NAME

Test no.	12
Input	Run the program by choosing different player names
Expected output	The program should run and ask a user to choose one player among three players

Actual	The program was run and ask a user to choose one player among three players
Test Result	Test successful

Table 12:Test 12 DIFFERENT PLAYER NAME

```

The list of best player of dear club of tarahara:
-----
| Code   | Player           |
-----
| LM     | Lionel Messi    |
-----
| NJ     | Neymar Junior   |
-----
| KC     | Kiran Chemjong  |
-----
| ZZ     | Zheng Zhi       |
-----
| HK     | Harry Kane      |
-----

#####
#      PLEASE!! CHOOSE THREE PLAYER      #
#####
LM NJ KC
1) LM
2) NJ
3) KC
PLEASE CHOOSE THREE PLAYERS AMONG FOLLOWING BY TYPING NUMBER:

```

Figure 12:test 12 DIFFERENT PLAYER NAME

3.0.13. Test 13 wrong user id.

Test no.	13
Input	Run the program by choosing the wrong player id
Expected output	The error message should show and give another chance to choose one player among three.
Actual	The error message was showed and give another chance to choose one player among three.

Test Result	Test successful
-------------	-----------------

Table 13:Test 13 wrong user id

```
#####
#      PLEASE!! CHOOSE THREE PLAYER      #
#####
LM NJ KC
1) LM
2) NJ
3) KC
PLEASE CHOOSE THREE PLAYERS AMONG FOLLOWING BY TYPING NUMBER:4
#####
#      ERROR!! PLEASE CHOOSE AMONG 1, 2, 3 NUMBERS ONLY      #
#####
PLEASE CHOOSE THREE PLAYERS AMONG FOLLOWING BY TYPING NUMBER:
```

Figure 13:Test 13 wrong user id

3.0.14. test 14 PROFILE NA BHAKO PLAYER ID.

Test no.	14
Input	Run the program by choosing unavailable player id
Expected output	the file is not available error message should show and give another chance to choose a player whose profile is available
Actual	the file is not available error message was showed and another chance was given to choose a player whose profile is available
Test Result	Test successful

Table 14:Test 14 PROFILE NA BHAKO PLAYER ID.

```
#####
# PLEASE!! CHOOSE THREE PLAYER #
#####
LM NJ ZZ
1) LM
2) NJ
3) ZZ
PLEASE CHOOSE THREE PLAYERS AMONG FOLLOWING BY TYPING NUMBER:3
#####
# ERROR!! THE FILE IS NOT AVILABLE #
#####
```

Figure 14:Test 14 PROFILE NA BHAKO PLAYER ID.

3.0.15. Test 15 right user-id

Test no.	15
Input	Run the program by choosing the right player id
Expected output	The program should run and ask a user to continue or exit
Actual	The program was run and ask a user to continue or exit
Test Result	Test successful

Table 15:Test 15 right user id

```
#####
# PLEASE!! CHOOSE THREE PLAYER #
#####
LM NJ KC
1) LM
2) NJ
3) KC
PLEASE CHOOSE THREE PLAYERS AMONG FOLLOWING BY TYPING NUMBER:1
Lionel Andrés Messi[note 1] (Spanish pronunciation: [ljoˈnel anˈdres ˈmesi] ( listen);
[A] born 24 June 1987) is an Argentine professional footballer who plays as a forward
and captains both Spanish club Barcelona and the Argentina national team. Often
considered as the best player in the world and widely regarded as one of the greatest
players of all time, Messi has won a record six Ballon d'Or awards, a record
six European Golden Shoes, and in 2020 was named to the Ballon d'Or Dream Team.
#####
# PRESS y/Y TO CONTINUE #
#####
```

Figure 15:Test 15 right user id

3.0.16. Test 16 EXIT YES

Test no.	16
Input	Exit the program
Expected output	The program should exit showing the message thank you
Actual	The program was exited showing message thank you
Test Result	Test successful

Table 16:Test 16 EXIT YES

```

#####
#      PRESS y/Y TO CONTINUE      #
#####
N
##### ##      ##      ##      ##      ##      ##      ##      ##      ##      ##
##      ##      ##      ##      ##      ##      ##      ##      ##      ##      ##
##      ##      ##      ##      ##      ##      ##      ##      ##      ##      ##
##      ##### ##      ##      ##      ##      ##      ##      ##      ##      ##
##      ##      ##      ##### ##      ##      ##      ##      ##      ##      ##
##      ##      ##      ##      ##      ##      ##      ##      ##      ##      ##
##      ##      ##      ##      ##      ##      ##      ##      ##      ##      ##
(sujeanc@ LAPTOP-D0DIEB8C) - [~]
$

```

Figure 16:Test 16 EXIT YES

3.0.17. Test 17 EXIT NO

Test no.	17
Input	continue the program
Expected output	The program should continue and ask to select country
Actual	The program was continuing and ask to select country

Test Result	Test successful
-------------	-----------------

Table 17: Test 17 EXIT NO

```
#####
#      PRESS y/Y TO CONTINUE      #
#####
Y
Please choose the best football team
from following country?
-----
| Code      | Country    |
-----
| BRZ       | Brazil     |
-----
| ARG       | Argentina  |
-----
| NEP       | Nepal      |
-----
| CHI       | China      |
-----
| ENG       | England    |
-----
```

Figure 17: Test 17 EXIT NO

4.0. Contents of three files: (TEXTS) 6

- **LM**

Lionel Andrés Messi^[note 1] (Spanish pronunciation: [ljoˈnel anˈdres ˈmesi] (listen); [A] born 24 June 1987) is an Argentine professional footballer who plays as a forward and captains both Spanish club Barcelona and the Argentina national team. Often considered as the best player in the world and widely regarded as one of the greatest players of all time, Messi has won a record six Ballon d'Or awards, a record six European Golden Shoes, and in 2020 was named to the Ballon d'Or Dream Team.

- **NJ**

Neymar da Silva Santos Júnior (Brazilian Portuguese: [nejˈmaʁ de ˈsiwvɐ ˈsɛtus ˈʒũɲoʁ]; 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. Neymar came into prominence at Santos, where he made his professional debut aged 17.

- **KC**

Born in Dhankuta, Chemjong graduated from the ANFA Academy and joined the Machhindra Football Club. After spending one year at the club Chemjong's impressive performances earned him a move to the Three Star Club in 2007. He has won British Gurkha Cup and Aaha Gold Cup Football Tournament.

5.0. Conclusion

As an overall conclusion, It was a very wonderful and learning experience for me while working on this project. The joy of work and the thrill involved while tackling the various problems and challenges feels very satisfying. While coding many difficulties were faced. After watching many videos and reading articles and with the help of friends and tutor coding was done. By creating functions to run the code. Use of while loop, if-else, elif are used in the coding section. The validation is completed by giving a very instructive error message to execute the program without any problems. The design is made user-friendly for the better experience of the user.

A fully executable program is programmed in a bash shell. The coding is done by fulfilling the demand of the coursework. To run the program user, need to give a name and id as an argument and after passing the argument secret key is asked for entrance. While validating the program according to the testing scenario many errors occur while solving the error new knowledge was gained. Learned many new command-line, shortcuts in Kali Linux as a windows operating system user Linux was hard at the beginning stage but by the time using it time to time get easier to use. By doing proper research the project is completed successfully.

Task B**6.0. Introduction**

There are many basic problems in memory managing is knowing when to keep the data it contains and when to throw it for the reuse of memory again. This may sound minor problem but in fact, it is such a hard problem that it is an entire field of study in its own right. Most programmers wouldn't have to think about memory management in a perfect world. Unfortunately, bad memory management practices can have a variety of effects on program robustness and speed, both in manual and automated memory management.

The common issues of memory management are many programs release memory, but when they try to attempt it later, they crash or act erratically. The surviving reference to the memory is known as a hanging pointer, and the state is known as a premature free. The majority of the time, this is limited to manual memory management. Some programs allocate memory indefinitely without ever giving it up, resulting in a run out of memory. A memory leak is a term for this situation. A poor allocator will do such a bad job of giving out and receiving memory blocks that it can no longer allocate large enough blocks despite having enough spare memory. This is because free memory can be divided into several small blocks, separated by blocks still in use. This condition is known as external fragmentation. Another problem with the layout of allocated blocks comes from the way that modern hardware and operating system memory managers handle memory. Memory managers may also trigger serious performance issues if they were configured for one purpose but are used for another.

(Alain Azagury, 2020)

7.0. Aims and objectives

It decides which should reside in the main memory. Manages the non-core resident sections of a process' virtual address space. Monitors the available main memory and writes processes to the swap system on a regular basis to allow more processes to run in the main memory at the same time. Caching, at multiple levels. Virtual address 0 is allocated to the first address of the process. Ensures security and privacy of data & instructions. Operating System allows data sharing ensure with protection at a time.

8.0. Background

8.1.1. Physical memory

Physical memory (also known as random access memory (RAM)) is a form of very fast, but volatile data storage. RAM modules are always measured in nanoseconds (1000^{-3}), and physical disks are always measured in milliseconds (1000^{-1}). This makes physical memory forcibly 100,000 times faster than a common physical disk. As a result, Windows and Windows Server retain the most frequently accessed pages of memory in physical memory wherever possible, relying on a disk only when necessary.

When a system has low space on physical memory, it often leads to the delays of system, in the worst scenario, a complete lag of the system. It is attached to the motherboard of the system. Primary memory is the only storage type that is straightforwardly addressable to the CPU. Primary memory provides the instructions of the program to execute. RAM is needed when the computer requires memory usage to run the application program or to open files it is the first memory to be getting used.

Physical memory consumes a consequential amount of electricity, so using less amount of physical memory will consume less power which decreases the power cost. The best way to see how much actual memory is “free”—not to be mistaken with “available” memory—is to see how much memory is “available.” Free memory is the sum of the free page list and the zero-page list. A free page list is a list of physical memory pages that need to be cleaned, while a zero page list is a list of cleaned physical memory pages. Simply put, the more free memory you have, the more you can delete without affecting device output.

(Huffman, 2015)

(Prerana Jain, January 03, 2019)

8.1.2. Memory placement

It is primarily a computer hardware operation but it is handled through the software application and operating system. In both physical and virtual memory management, the memory allocation process is very similar. When programs and services are run, they are given a specific amount of memory based on their needs. The memory is released and transferred to another program or combined within the primary memory until the program has completed its operation or is idle.

It has two core types

- Static memory allocation

When the compiler compiles the program and generates object files, the linker merges all of these object files into a single executable file, and the loader loads this single executable file into main memory for execution, static memory allocation occurs.

- Dynamic memory allocation

While the program is in execution dynamic memory allocation is performed. When the program's entities are used for the first time while the program is running, memory is allocated.

(T, 2019)

There are different placement algorithms

- First fit

The partition is allocated in the first block which has sufficient space from the top of the main memory. It scans first and then searches for the first

available block which is large enough. It allocates the first hole that is large enough.



Figure 18: first fit

- Best fit

In best fit, the first smallest sufficient partition among the free available partition is allocated in a process. It seeks for the smallest hole from the list of holes whose size is greater or equal to the size of the process.

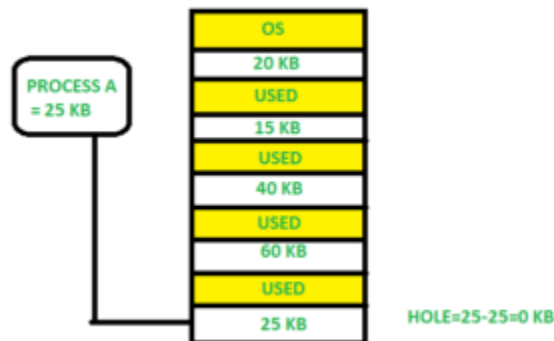


Figure 19: best fit

- Worst fit

In worst fit the largest sufficient among the freely available partitions available in the main memory is allocated in a process. It is just opposite to the best-fit algorithm. It seeks the largest hole from the entire list of holes and allocates it to process.



Figure 20:worst fit

- Next fit

It is similar to the first fit but the difference is its search for the first sufficient partition from the last allocation point and in the first fit first sufficient partition search from the top allocation point.

(vaivhavRai, et al., 06 Nov, 2020)

(T, 2019)

8.1.3. Page coloring

Page coloring is a performance enhancement that makes the most of the processor cache when viewing contiguous pages in virtual memory. Back to 10+ years ago, processor caches tended to map virtual memory rather than physical memory. This resulted in many issues, including the need to clear the cache on any context switch in some cases and data aliasing issues in the cache. Modern processor caches specifically map physical memory to solve these issues. This means that two adjacent pages in a process's address space might not correspond to two adjacent pages in the cache. If you're not careful, side-by-side virtual memory pages can end up using the same page in the processor cache, causing cacheable data to be discarded prematurely and lowering CPU performance.

The mechanism of page coloring is to control partitioning of cache, and also consequently to achieve fair and utilization of cache efficient amount. However, page coloring is the one that places additional constraints on memory space allocation, which may conflict with application memory needs. Furthermore, adaptive cache partitioning policies in a multi-programmed execution environment can result in significant page recoloring overhead (or copying). This paper proposes a hot-page coloring method for each process that involves coloring only on a small number of regularly accessed (or hot) pages. By leveraging spatial locality information during a page table scan of access bits, the cost of finding hot pages online is reduced. Our results indicate that in practice, hot page recognition and selective coloring can substantially reduce the coloring-induced negative effects.

(Eom, 2011)

(dillon & dillon, 2000)

8.1.4. Paging

Paging is an approach in which the main memory of a computer system is arranged in the form of equal-sized blocks called pages. In this technique, the address of occupied pages of physical memory is stored in a table, which is known as a page table.

Paging empowers the operating system to obtain data from the physical memory location without determining lengthy memory addresses in the instruction. In this technique, while mapping the physical address of the data the virtual address is used. The length of the virtual address is specified in the instruction and is smaller than the physical address of the data. It consists of two different numbers, the first number is the address of the page called virtual page in the page table and the second number is the offset value of the actual data on the page.

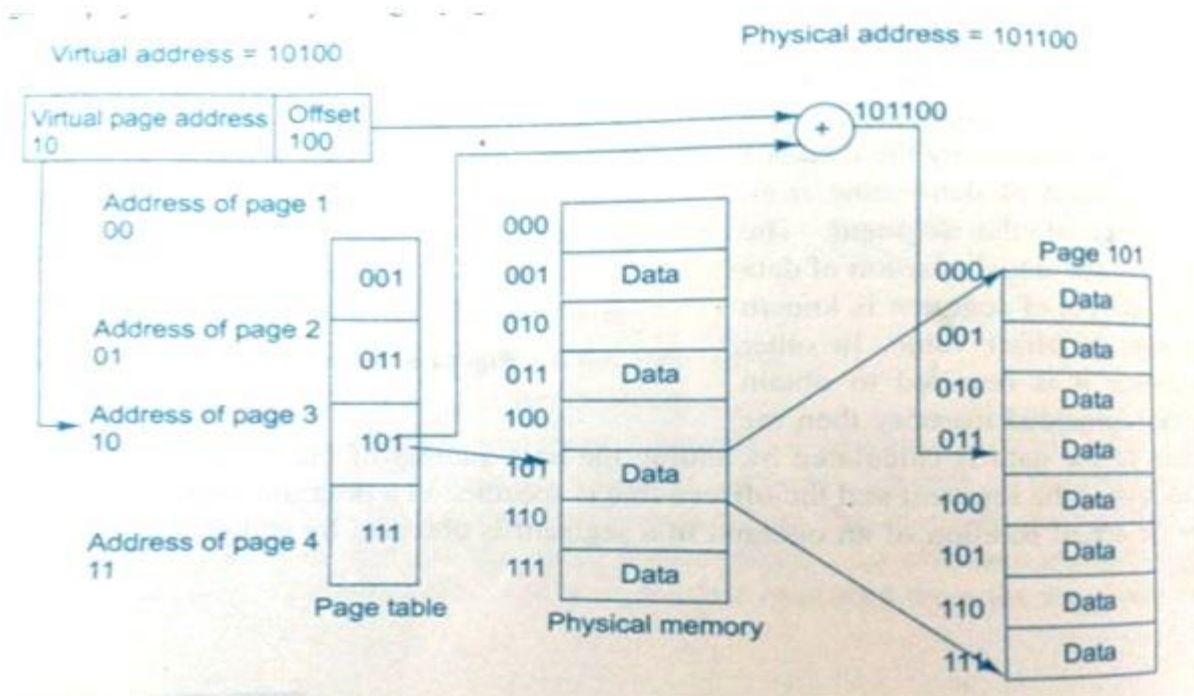


Figure 21: paging

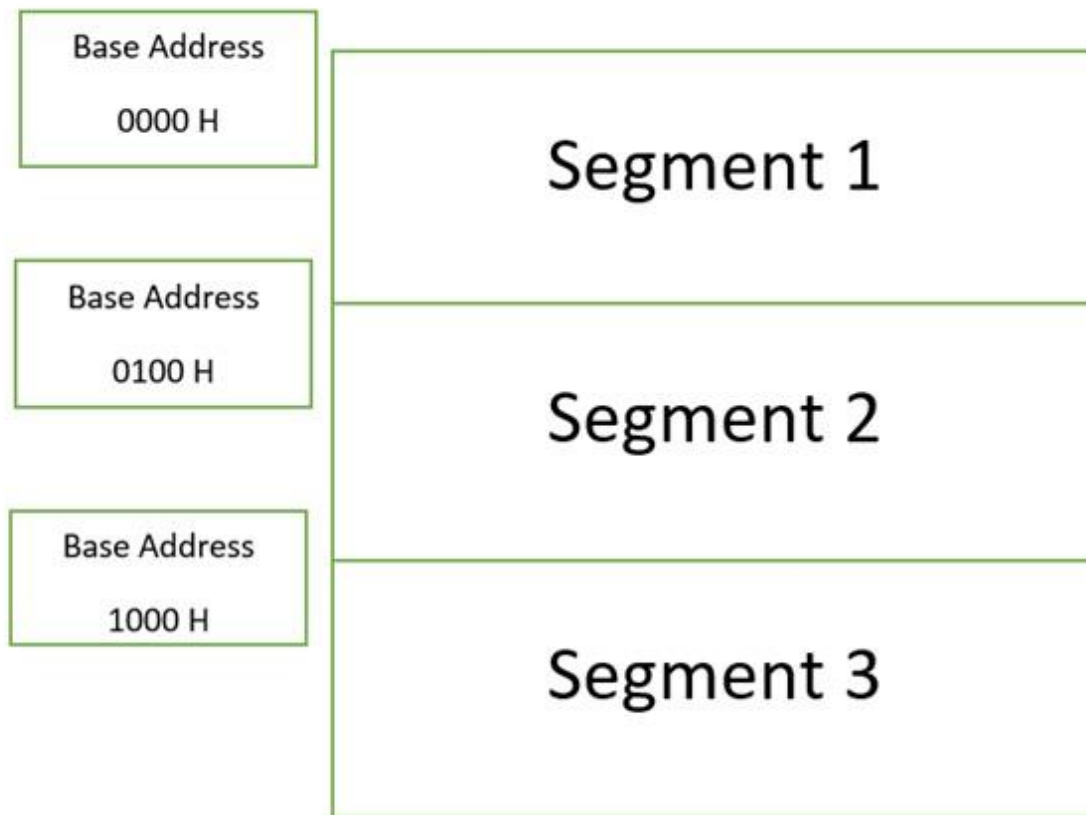
courtesy: Fundamental of computers by E. Balagurusamy

The above figure shows how the virtual address is used to get the physical address of an occupied page of physical memory using a page table.

(Amit & shulkla, 2017)

8.1.5. Segmentation

Segmentation refers to the approach of dividing the physical memory space into multiple blocks. Each block of memory has a specific length and is known as a segment. Every segment has an own starting address called the base address. The length of the segment determines the availability of memory space in the segment.



The above figure shows the organization of segments in a memory unit.

The distance between the actual position of the data value and the base address of the segment can be used to determine the location of data values stored in the segment. The displacement or offset value is the distance between the actual position of data and the segment's base address. In other words, when data must be retrieved from a segmented memory, the actual address of data is calculated by adding the segment's base address by an offset value. The segment's base address and offset value are specified in the program instruction itself.

The following diagram illustrates how the base address and offset value are added to get the actual position of an operand in a segment.

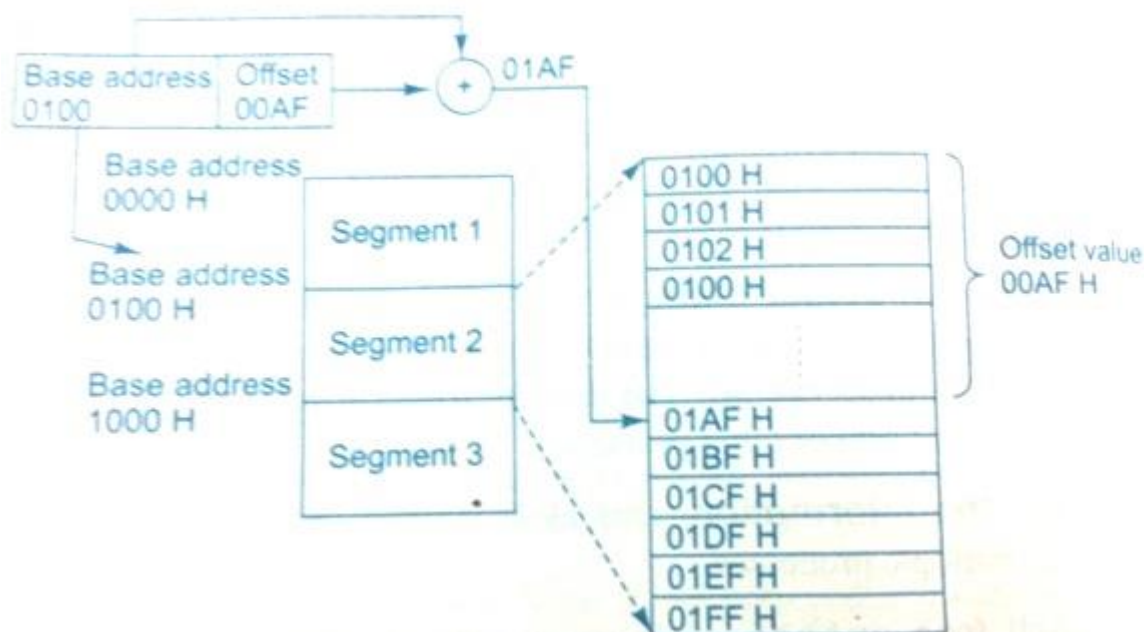


Figure 22:segmentation

courtesy: Fundamental of computers by E. Balagurusamy

8.1.6. Page size variation

In computers, .page size means the size of the page, which is a block of stored memory.

Different page sizes in all architectures: When running programs, the page size has an impact on the amount of memory and space required. When a program starts, most operating systems decide the page size. This function allows it to measure the most effective memory usage when the program is running.

On the web page, the page is known as page weight. It refers to the overall size of the webpage. Many of the files that create a web page are included in the page size. The HTML text, and included photos, style sheets, scripts, and other media are all included in these files.

Architecture	Smallest page size	Larger page size
32-bit x86 ^[18]	4 KiB	4 MiB in PSE mode, 2 MiB in PAE mode ^[19]

64-bit x86 ^[18]	4 KiB	2MiB, 1 GiB(only when the CPU has PDPE1GB flag)
IA-64(Itanium) ^[20]	4 KiB	8 KiB, 64 KiB, 256 KiB, 1 MiB, 4 MiB, 16 MiB, 256 MiB
Power ISA[21]	4 KiB	64 kiB, 16 MiB, 16 GiB
SPARC v8 with SPARC Reference MMU[22]	4 KiB	256 KiB, 16 MiB
UltraSPARC Architecture 2007[23]	4 KiB	64 KiB, 512 KiB (optional), 4 MiB, 32 MiB (optional), 256 MiB (optional), 2 GiB (optional), 16 GiB (optional)
ARMv7[24]	4 KiB	64 KiB, 1 MiB ("section"), 16 MiB ("supersection") (defined by a particular implementation)

(computer hope, 2017)

(Kruegel & Kruegel , 2016)

9.0. Conclusion

As an overall conclusion, It was a very wonderful and learning experience for me while working on this project. This allows me to get a deep knowledge of Memory placement, paging, and segmentation, physical Memory while researching for this project, and also increase my researching skills. Many difficulties were faced and tackled by the time. While researching for answers for the report It was very hard to find because the content of project work was not available easily after reading many PDFs, articles, eBooks, etc.

The field of memory management is vast, complicated, time-consuming to study, and difficult to put into practice. This is partly due to the complexity of modeling how real multi-programmed systems behave. Page replacement policies, which have been the subject of a lot of studies, are a good example because they've only ever been proven to work well for specific workloads. Administrators tune systems as well as analysis and algorithms development to solve the issue of adjusting algorithms and policies to various workloads. Paging is a storage mechanism that allows the operating system to retrieve processes from secondary storage and store them as pages in the main memory. The only difference between segmentation and paging is that segments have variable length, while pages in the paging system are usually of a fixed size. Memory management algorithm of segmentation very costly. Based on measurements of program behavior and basic models of system behavior, the impact page size can have on various system performance measures is analyzed.

10.0. References

Alain Azagury, E. K. K. E. P. Z. Y., 2020. *Introduction to memory management*, s.l.: s.n.

Amit & Shukla, 2017. Memory Management Techniques in Operating System.

C. & Kruegel, 2016. *Operating system*. s.l.:UC Santa Barbara.

computer hope, 2017. *Computer Hope*. [Online]
Available at: <https://www.computerhope.com/jargon/p/pagesize.htm>
[Accessed 9 6 2021].

Eom, J. K. K. A. I., 2011. *Computational Science and Its Applications - ICCSA*. LNCS, volume 6784 ed. s.l.:Springer, Berlin, Heidelberg.

Huffman, C., 2015. physical memory. In: *Windows Performance Analysis Field Guide*. s.l.:Syngress, pp. 199-235.

M. & Dillon, 2000. Design elements of the FreeBSD VM system. *FreeBSD*.

Prerana Jain, January 03, 2019. operating system. *Concept of physical and virtual memory in Operating System*.

T, N., 2019. *binaryterm*. [Online]
Available at: <https://binaryterms.com/contiguous-memory-allocation-in-operating-system.html>
[Accessed 9 April 2021].

T, N., 2019. *BinaryTerm*. [Online]
Available at: <https://binaryterms.com/static-and-dynamic-memory-allocation.html>
[Accessed 9 April 2021].

v., h. & d., 06 Nov, 2020. geeksforgeeks. *operating system*.

11.0. Appendix

- Memory Controller

While operating, computers store all active data in Random Access Memory (RAM) chips. RAM chips are connected in motherboard of computer, and linked to the computer's processor through the front side bus. They act as a kind of direct highway for the transfer of variables and program data. The memory controller is a chip that is usually found on the motherboard's northbridge. It manages read and write operations with system memory, as well as keeping the RAM active by providing electric current to the memory. Other types of storage, such as hard drives and optical discs, are generally slower than RAM. However, one of RAM's drawbacks is that it requires a constant supply of power in order to function. The information stored in RAM chips is lost as soon as the power supply is cut off. This requirement is fulfilled by the memory controller, which "refreshes" the RAM at a constant rate while the com

- Virtual Memory

Virtual Memory is a place where very heavy programs can be stored in form of pages while their execution and only the required pages or processes are loaded into main memory in chunks. This technique is useful as it provides a large virtual memory for user programs when the physical memory is small. As a result, virtual memory is a technique that enables the execution of processes that are not entirely stored in physical memory. Virtual Memory mainly gives the illusion of more physical memory than there really is with the help of Demand Paging.

The software need memory space virtual memory gives permission to run (i.e., a logical memory) whose size is greater than the computer's RAM. Most PCs sold in this day contains from 256MB to 1024MB of RAM. While this is significant in comparison to what was common only a few years ago, it is commonly insufficient to run all of the programs that users attempt to run simultaneously. The reason is that the size of many programs has continued to increase accompanying the growth in memory sizes and HDD capacities, largely in order to add more features (including fancier graphics)

- Cache Memory

Cache memory is of high-speed type random access memory (RAM) which is pre-installed on the processor. This allows for faster processing as the processor does not have to wait for the data and instructions to be fetched from RAM. Data can be transferred to and from

cache memory much faster than from RAM. As a result, cache memory is used to store data temporarily and instructions that are likely to be reused by the processor. The computer runs more faster if it has more cache memory. On other hand, cache memory is expensive to manufacture than RAM because of its high speed performance.