



 slington college

(इस्लिङ्टन कलेज)

**Module Code & Module Title**

**CS5001NI Networks and Operating System**

**Assessment Weightage & Type**

**20% Individual Coursework**

**Year and Semester**

**2021-22 Autumn**

**Student Name: Rishabh Singh**

**London Met ID: 20048981**

**College ID: NP01CP4S210251**

**Assignment Due Date: 25<sup>th</sup> April 2022**

**Assignment Submission Date: 25<sup>th</sup> April 2022**

**Title: Unix Script Programming (Task A)**

**Process Management (Task B)**

**Word Count (Task B): 2022**

*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
Introduction .....	1
Script .....	2
Testing .....	8
Test 1:.....	8
Test 2:.....	9
Test 3:.....	10
Test 4:.....	11
Test 5:.....	12
Test 6:.....	13
Test 7:.....	14
Test 8:.....	15
Test 9:.....	16
Test 10:.....	17
Test 11:.....	18
Test 12:.....	19
Test 13:.....	20
Test 14:.....	21
Test 15:.....	22
Test 16:.....	23
Contents of three files: (TEXTS) .....	24
• JL .....	24
• FM .....	24
• KC .....	24
Conclusion .....	25
Task B .....	26
Introduction .....	26
Aims and Objectives.....	27
Background: .....	27

Process Architecture.....	27
Process Control Blocks.....	29
Process States .....	31
Process Hierarchies.....	34
Implementation of process.....	35
Conclusion .....	36
References.....	37
Appendix .....	39
Appendix – A (Glossary).....	39
Appendix – B (Process Scheduling) .....	39
Appendix – c (Priority Scheduling).....	40

## List of Figures

Figure 1: To run the script without username .....	8
Figure 2: Run the script with username and id .....	9
Figure 3: To run the incorrect password 3 times .....	10
Figure 4: Run correct password .....	11
Figure 5: Entering band name instead of code.....	12
Figure 6: Incorrect band code .....	13
Figure 7: Enter correct band code .....	14
Figure 8: Pick same member twice .....	15
Figure 9: Pick non existing band member name. ....	16
Figure 10: Picking 4 member names.....	17
Figure 11: To pick a member with non-existing file .....	18
Figure 12: select existing external file .....	19
Figure 13: select existing external file .....	20
Figure 14: Selecting no when asked to start over program. ....	21
Figure 15: Entering wrong id (parameter validation) .....	22
Figure 16: Entering non existing number in member file selection .....	23
Figure 17: Process architecture (Sites Bay, 2020) .....	27
Figure 18: Process control block (PCB) (Williams, 2022).....	29
Figure 19: Process States (Williams, 2022) .....	31
Figure 20: Process Hierarchies (Anon., 2002) .....	34
Figure 21: Process table (Athiralekshmicv, 2014) .....	35
Figure 22: Process Scheduling .....	39
Figure 23: Priority Scheduling (Data Flair, 2022) .....	40

## List of Tables

Table 1: Test 1 .....	8
Table 2: Test 2 .....	9
Table 3: Test 3 .....	10
Table 4: Test 4 .....	11
Table 5: Test 5 .....	12
Table 6: Test 6 .....	13
Table 7: Test 7 .....	14
Table 8: Test 8 .....	15
Table 9: Test 9 .....	16
Table 10: Test 10 .....	17
Table 11: Test 11 .....	18
Table 12: Test 12 .....	19
Table 13: Test 13 .....	20
Table 14: Test 14 .....	21
Table 15: Test 15 .....	22
Table 16: Test 16 .....	23

## **Task A**

### **Introduction**

In this part of the coursework a script file is made keeping the scenario provided in mind and working accordingly. In order to complete the task many programming methods were used like loops, if then else, function, select statement, case, while loop and until loop. The shell scripting used in this coursework is bash (Borne Again Shell) it's the free and improved version of Borne shell that is distributed with Linux and GNU operating system. Bash is very compatible with sh and has some useful features from Korn shell and C shell (Free Software Foundation, Inc., 2008). The complete process for this task is in the document below. A proper test was carried out to check the functionality of the script in real world scenario.

## Script

```
#!/bin/bash

# To check if the user entered 2 parameters
if [ $# != 2 ]
then

    echo -e "SORRY! could not enter the program"
    echo
    sleep 0.6
    echo -e "Error! Username or ID missing"
    echo
    sleep 0.6
    echo -e "Instruction bash <programname> <username> <ID>"
    echo
    sleep 1
    echo -e "\e[31mTerminating program \e[0m"
    sleep 1
    echo -e ".\c"
    sleep 1
    echo -e ".\c"
    sleep 1
    echo -e ".\c"
    sleep 1
    echo -e ".\c"
    sleep 1
    echo "."
    sleep 1
    echo
    echo "Program terminated successfully."
    exit 1
fi

#To check if the first parameter is username and second parameter is ID
until [[ "$1" =~ ^[A-Za-z]+$ && "$2" =~ ^[+-]?[0-9]+$ ]]
do
    sleep 1
    echo
    echo " INVALID PARAMETER "
    sleep 1
    echo
    echo " Error! Enter Username as letter and ID as a number "
    exit
done
```

```
figlet "WELCOME"

# Function to ask user for secret key
input_password(){
echo -e "Enter the secretkey: \c"
read -s key
}

input_password
secretkey=test123
tries=1

# Loop for checking is the secretkey is valid or not
while [ $tries -lt 3 ]
do
    if [ $secretkey = $key ]
    then
        echo ""
        echo -e "Loading..\c"
        echo -e ".....Loading 100%" | pv -qL 10
        sleep 1
        figlet "Hi"
        sleep 1
        echo
        echo "=====
        echo "      WELCOME TO THE PROGRAM!"
        echo
        echo " ID: "$2" "
        echo " Name: "$1" "
        echo " Date: $(date +%m/%d/%Y) "
        echo " Time: $(date +%r) "
        echo
        echo "=====
        sleep 1
        break
    else
        echo -e "Error! Invalid Secret Key please try again"
        input_password
        tries=`expr $tries + 1`
    fi
done

# Loop for terminating the program after entering wrong password for t
he third time.
while [ $tries = 3 ]
do
```

```

echo "Sorry! You entered wrong password for 3 times"
echo
echo -e "\e[31m Terminating program \e[0m"
sleep 0.7
echo -e ".\c"
sleep 0.7
echo -e ".\c"
sleep 0.7
echo -e ".\c"
sleep 0.7
echo -e ".\c"
sleep 0.7
echo "."
sleep 0.7
echo
echo "Program terminated successfully."
exit 1
done

# Function to ask user for the best band
band_selection(){
    echo -e "\n\t\t SELECT A BAND"
    echo -e "
    echo -e "| Band | code |"
    echo -e "-----"
    echo -e "| Beatles | BEA |"
    echo -e "| AC/DC | AD |"
    echo -e "| Queen | QUE |"
    echo -e "| Blondie | BLO |"
    echo -e "| Nirvana | NIR |"
    echo -e "
    sleep 1
    echo -e "\nWhich is the best music band(Enter band code):-\c"
    read best_band_code
    if [[ "$best_band_code" = "AD" ]]; then
        echo -e "Congrats! you choose the best band"
        echo -e "
e "=====
=====
        sleep 1
        echo -e "\t\t\t AC/DC is the best band"
        sleep 1
        echo -e "The Scottish-
born brothers Malcolm and Angus Young founded the Australian"
        sleep 1

```



```

    echo -
e "rock band AC/DC in Sydney in 1973. Hard rock, blues rock and heavy
metal"
    sleep 1
    echo -
e "have all been used to describe the band's music; however, rock and
roll is what"
    sleep 1
    echo -e "they call it."
    echo -
e "=====
=====
    elif [[ "$best_band_code" = "BEA" || "$best_band_code" = "QUE" || "$
best_band_code" = "BLO" || "$best_band_code" = "NIR" ]]; then
    sleep 1
    echo -e "Error! Wrong Band!"
        sleep 1
        echo -e "$best_band_code is not the best music band"
        band_selection
    else
        echo -e "Error! Invalid Input!"
        band_selection
    fi
}

member_selection(){
    echo -e "\nThe Five Star band members are:"
    echo -e "
    echo -e "| Members                | code                |"
    echo -e "-----"
    echo -e "| John Lennon            | JL                  |"
    echo -e "| Angus Young            | AY                  |"
    echo -e "| Freddie Mercury       | FM                  |"
    echo -e "| Debbie Harry           | DH                  |"
    echo -e "| Kurt Cobain            | KC                  |"
    echo -e "
    sleep 0.7
    echo
    echo -
e "Choose three band members represented by codes separated by spaces:
"
    read a b c
    number_validate $a $b $c
    validate "$a"
    validate "$b"
    validate "$c"

```

```
}
validate(){
    # to check if the players are valid
    if [[ $1 != "JL" && $1 != "AY" && $1 != "FM" && $1 != "DH" && $1 !=
"KC" ]]; then
        sleep 0.7
        echo -e "Error! band member doesnt exist Try again"
        sleep 0.7
        member_selection
    fi
}
number_validate(){
    # to check if there are exactly 3 inputs
    if [[ $# != 3 ]]; then
        sleep 0.7
        echo -e "Error! Invalid number of inputs, Enter only 3 band members"
        sleep 0.7
        member_selection
    fi
    # to check if there are duplicate inputs
    if [[ "$1" == "$2" || "$2" == "$3" || "$1" == "$3" ]]; then
        sleep 0.7
        echo -e "Error! Duplicate inputs!"
        sleep 0.7
        member_selection
    fi
}
# Function for choosing file
file(){
    PS3="Enter a number as per list for choosing a band member you want:
"
    select member in $a $b $c
    do
        if [ -z $member ]
        then
            echo -e "Please use correct number"
        elif [ -r $member ]
        then
            echo ""
            sleep 0.7
            echo -e "You selected" $member
            echo ""
            sleep 0.7
            cat $member
            break
        fi
    done
}
```

```
        else
            echo -e "Error! Can not find file!"
        band_selection
    fi
done
}

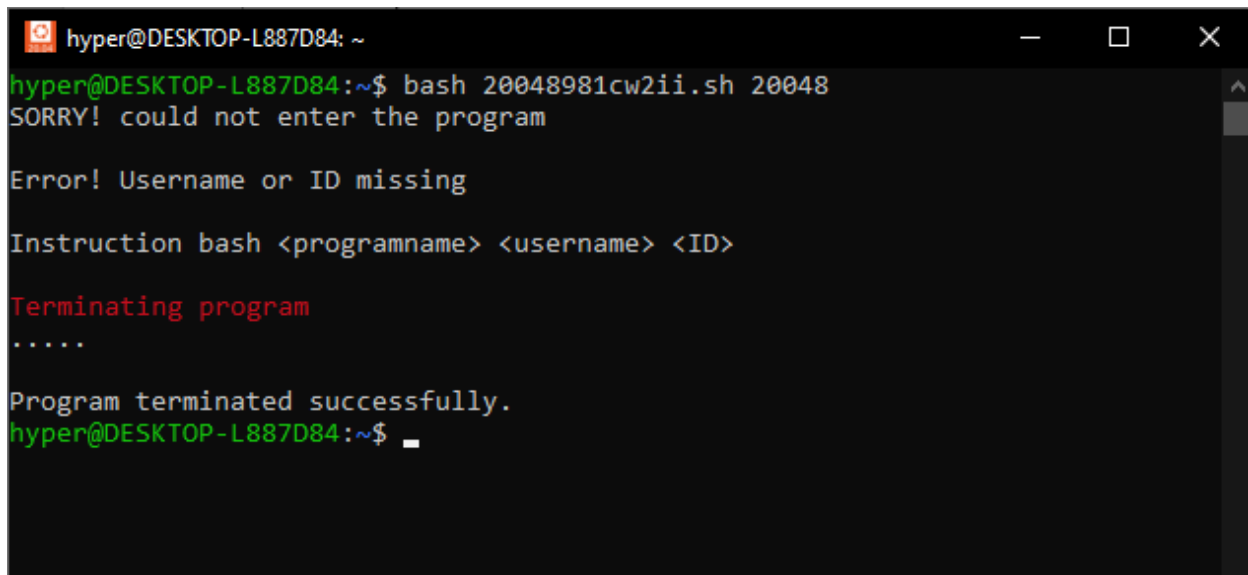
# Funtion to exit or restart the program
execute(){
band_selection
member_selection
file
echo -e "Do you want to start over?(y/n):-\c"
read y
case $y in
    "YES"|"yes"|"Yes"|"Y"|"y")
        execute
        ;;
    *)
        figlet Program Terminated Successfully
        exit
        ;;
esac
}
execute;
```

## Testing

**Test 1:** To run the script without username

Test No.	1
Objective	To run the script without username
Input	Opened the script file. <ul style="list-style-type: none"> <li>Entered only id</li> </ul> Argument: bash 20048981cw2ii.sh 20048
Expected Output	Shows proper error message gives instruction and terminates the program.
Actual Output	The program gave user an error message gave instruction how to enter details and terminated the program
Test Result	Successful

Table 1: Test 1



```

hyper@DESKTOP-L887D84: ~
hyper@DESKTOP-L887D84:~$ bash 20048981cw2ii.sh 20048
SORRY! could not enter the program

Error! Username or ID missing

Instruction bash <programname> <username> <ID>

Terminating program
.....

Program terminated successfully.
hyper@DESKTOP-L887D84:~$

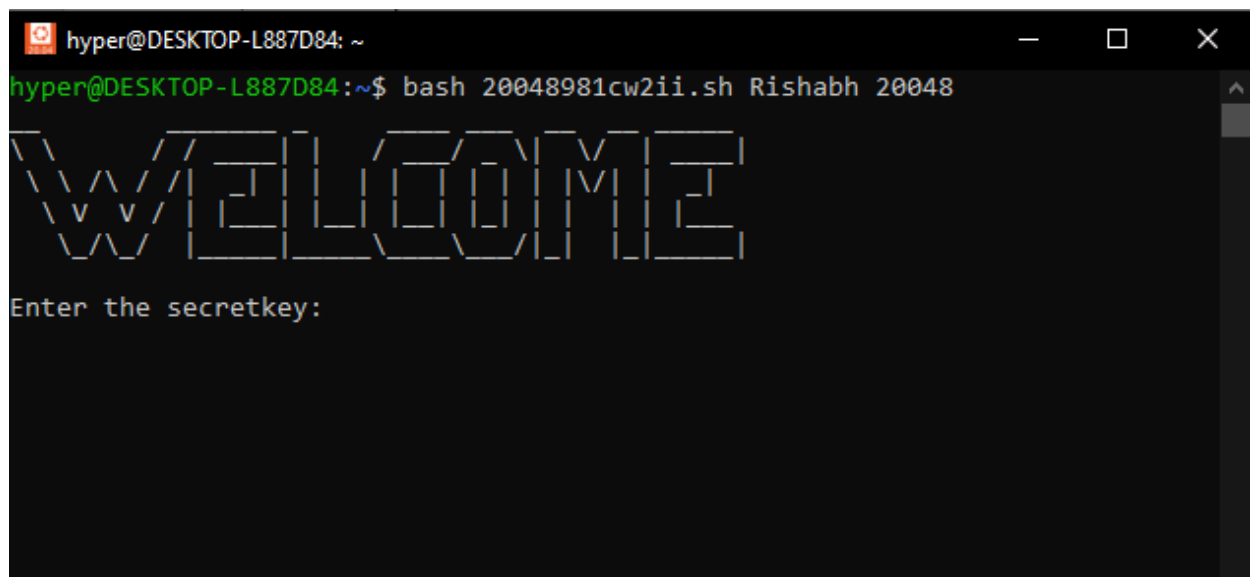
```

Figure 1: To run the script without username

**Test 2:** Run the script with username and id

Test No.	2
Objective	Run the script with username and id
Input	Opened the script file <ul style="list-style-type: none"><li>Entered both username and id</li></ul> Argument: bash 20048981cw2ii.sh Rishabh 20048
Expected Output	The program gets executed a welcome message will appear and ask user for secret key.
Actual Output	The program got executed welcome message was displayed and secret key was asked.
Test Result	Successful

Table 2: Test 2



```
hyper@DESKTOP-L887D84: ~  
hyper@DESKTOP-L887D84:~$ bash 20048981cw2ii.sh Rishabh 20048  
WELCOME  
Enter the secretkey:
```

Figure 2: Run the script with username and id

**Test 3:** To run the incorrect password 3 times

Test No.	3
Objective	To run incorrect password 3 times
Input	<p>Opened the script file</p> <ul style="list-style-type: none"> <li>Entered both username and ID</li> <li>Entered wrong password 3 times</li> </ul> <p>Argument bash 20048981cw2ii.sh Rishabh 20048</p>
Expected Output	After using the incorrect key, it should prompt to enter the correct key for 3 times then exit the program.
Actual Output	When the user enters incorrect key the program requests to enter the correct key if the key is wrong for 3 times the program gets terminated.
Test Result	Successful

Table 3: Test 3

```

hyper@DESKTOP-L887D84: ~
hyper@DESKTOP-L887D84:~$ bash 20048981cw2ii.sh Rishabh 20048

WELCOME

Enter the secretkey: Error! Invalid Secret Key please try again
Enter the secretkey: Error! Invalid Secret Key please try again
Enter the secretkey: Sorry! You entered wrong password for 3 times

Terminating program
...
Program terminated successfully.
hyper@DESKTOP-L887D84:~$

```

Figure 3: To run the incorrect password 3 times

**Test 4:** Run correct password

Test No.	4
Objective	To run correct password
Input	<p>Opened script file</p> <ul style="list-style-type: none"> <li>Entered proper username and id</li> <li>Entered password</li> </ul> <p>Argument</p> <p>bash 20048981cw2ii.sh Rishabh 20048</p> <p>test123</p>
Expected Output	After the user enters secret key the program should welcome the user displaying id, name, date and time.
Actual Output	When the user enters correct password the program welcomes the user and displays user id, name, date and time.
Test Result	Successful

Table 4: Test 4

```

hyper@DESKTOP-L887D84: ~
hyper@DESKTOP-L887D84:~$ bash 20048981cw2ii.sh Rishabh 20048

WELCOME

Enter the secretkey:
Loading.....Loading 100%

HI

=====
WELCOME TO THE PROGRAM!

ID: 20048
Name: Rishabh
Date: 04/24/2022
Time: 11:00:35 PM

=====

SELECT A BAND

| Band | code |
|-----|-----|
| Beatles | BEA |
| AC/DC | AD |
| Queen | QUE |
| Blondie | BLO |
| Nirvana | NIR |

Which is the best music band:-

```

Figure 4: Run correct password

**Test 5:** Entering band name instead of code

Test No.	5
Objective	To enter band name instead of code
Input	Opened the script file <ul style="list-style-type: none"> <li>Entered bands name instead of code</li> </ul> Argument Beatles
Expected Output	After the user inputs anything except of code the program notifies user with error message.
Actual Output	When the user entered band name the program notified user with error message.
Test Result	Successfull

Table 5: Test 5

```

hyper@DESKTOP-L887D84: ~
SELECT A BAND
| Band | code |
|-----|-----|
| Beatles | BEA |
| AC/DC | AD |
| Queen | QUE |
| Blondie | BLO |
| Nirvana | NIR |

Which is the best music band:-Beatles
Error! Invalid Input!

SELECT A BAND
| Band | code |
|-----|-----|
| Beatles | BEA |
| AC/DC | AD |
| Queen | QUE |
| Blondie | BLO |
| Nirvana | NIR |

Which is the best music band:-_

```

Figure 5: Entering band name instead of code



**Test 6:** Incorrect band code

Test No.	6
Objective	To enter incorrect band code
Input	Opened the script file. <ul style="list-style-type: none"> <li>Entered wrong band code</li> </ul> Argument QUE, BLO, ASD
Expected Output	After entering the wrong band code, the program warns about a mistake and the user guesses until the correct option is chosen.
Actual Output	When the user entered wrong code, the program warned user about mistake and kept giving another chance to guess.
Test Result	Successfull

Table 6: Test 6

```

hyper@DESKTOP-L887D84: ~
SELECT A BAND
| Band | code |
|-----|-----|
| Beatles | BEA |
| AC/DC | AD |
| Queen | QUE |
| Blondie | BLO |
| Nirvana | NIR |
|-----|-----|
Which is the best music band:-QUE
Error! Wrong Band!
QUE is not the best music band

SELECT A BAND
| Band | code |
|-----|-----|
| Beatles | BEA |
| AC/DC | AD |
| Queen | QUE |
| Blondie | BLO |
| Nirvana | NIR |
|-----|-----|
Which is the best music band:-BLO
Error! Wrong Band!
BLO is not the best music band

SELECT A BAND
| Band | code |
|-----|-----|
| Beatles | BEA |
| AC/DC | AD |
| Queen | QUE |
| Blondie | BLO |
| Nirvana | NIR |
|-----|-----|
ASD
Which is the best music band:-ASD
Error! Invalid Input!

```

Figure 6: Incorrect band code

**Test 7:** Enter correct band code

Test No.	7
Objective	To enter correct band code
Input	The script file was opened <ul style="list-style-type: none"> <li>Entered correct band code</li> </ul> Argument AD
Expected Output	If the option is correct the program informs about it and gives one two sentence explanation about it.
Actual Output	After the user entered the correct code, the user was notified, and explanation was given.
Test Result	Successful

Table 7: Test 7

```

hyper@DESKTOP-L887D84: ~
=====
                        SELECT A BAND
=====
| Band          | code |
|-----|-----|
| Beatles       | BEA  |
| AC/DC         | AD   |
| Queen        | QUE  |
| Blondie       | BLO  |
| Nirvana       | NIR  |
|-----|-----|

Which is the best music band:-AD
Congrats! you choose the best band
=====
                        AC/DC is the best band
=====
The Scottish-born brothers Malcolm and Angus Young founded the Australian
rock band AC/DC in Sydney in 1973. Hard rock, blues rock and heavy metal
have all been used to describe the band's music; however, rock and roll is what
they call it.
=====

The Five Star band members are:
=====
| Members       | code |
|-----|-----|
| John Lenon    | JL   |
| Angus Young   | AY   |
| Freddie Mercury | FM  |
| Debbie Harry  | DH   |
| Kurt Cobain   | KC   |
|-----|-----|

Choose three band members represented by codes separated by spaces:

```

Figure 7: Enter correct band code

**Test 8:**

Test No.	8
Objective	Pick the same member's name twice
Input	The script file was opened <ul style="list-style-type: none"> <li>• Duplicate data was entered.</li> </ul> Argument JL AY JL
Expected Output	The program notifies the user about wrong input and the user gets more guesses.
Actual Output	After the duplicate name was entered the program notifies user and gave more chances.
Test Result	Sucessfull

Table 8: Test 8

```

hyper@DESKTOP-L887D84: ~
====
The Five Star band members are:
| Members          | code |
|-----|
| John Lennon      | JL   |
| Angus Young      | AY   |
| Freddie Mercury | FM   |
| Debbie Harry     | DH   |
| Kurt Cobain      | KC   |
|-----|

Choose three band members represented by codes separated by spaces:
JL AY JL
Error! Duplicate inputs!

The Five Star band members are:
| Members          | code |
|-----|
| John Lennon      | JL   |
| Angus Young      | AY   |
| Freddie Mercury | FM   |
| Debbie Harry     | DH   |
| Kurt Cobain      | KC   |
|-----|

Choose three band members represented by codes separated by spaces:

```

Figure 8: Pick same member twice

**Test 9:** Pick non existing band member name.

Test No.	9
Objective	To enter non existing band member names.
Input	Script file was opened. <ul style="list-style-type: none"> <li>Non existing names was entered.</li> </ul> Argument AS DS FD
Expected Output	The program notifies user of wrong input and the program give smore guesses.
Actual Output	After the user entered wrong names, the user was notified and more guess was given
Test Result	Successful

Table 9: Test 9

```

hyper@DESKTOP-L887D84: ~
Error! Duplicate inputs!

The Five Star band members are:
| Members          | code |
|-----|
| John Lennon      | JL   |
| Angus Young      | AY   |
| Freddie Mercury | FM   |
| Debbie Harry     | DH   |
| Kurt Cobain       | KC   |
|-----|

Choose three band members represented by codes separated by spaces:
AS DS FD
Error! band member doesnt exist Try again

The Five Star band members are:
| Members          | code |
|-----|
| John Lennon      | JL   |
| Angus Young      | AY   |
| Freddie Mercury | FM   |
| Debbie Harry     | DH   |
| Kurt Cobain       | KC   |
|-----|

Choose three band members represented by codes separated by spaces:

```

Figure 9: Pick non existing band member name.

**Test 10:** Picking 4 member names

Test No.	10
Objective	To pick 4 member names
Input	Script file was opened <ul style="list-style-type: none"> <li>Entered 4-member code</li> </ul> Argument JL AY FM KC
Expected Output	If the code is repeated the user gets notified and given more chances.
Actual Output	After the user entered wrong code the script notified the user and gave more chances.
Test Result	Successful

Table 10: Test 10

```

hyper@DESKTOP-L887D84: ~
Error! band member doesnt exist Try again

The Five Star band members are:

| Members          | code |
|-----|
| John Lennon      | JL   |
| Angus Young      | AY   |
| Freddie Mercury | FM   |
| Debbie Harry     | DH   |
| Kurt Cobain      | KC   |

Choose three band members represented by codes separated by spaces:
JL AY FM KC
Error! Invalid number of inputs, Enter only 3 band members

The Five Star band members are:

| Members          | code |
|-----|
| John Lennon      | JL   |
| Angus Young      | AY   |
| Freddie Mercury | FM   |
| Debbie Harry     | DH   |
| Kurt Cobain      | KC   |

Choose three band members represented by codes separated by spaces:

```

Figure 10: Picking 4 member names

**Test 11:** To pick a member with non-existing file

Test No.	11
Objective	To pick a member with non-existing file
Input	Script file was opened <ul style="list-style-type: none"> <li>Non existing file name was entered.</li> </ul> Argument 3
Expected Output	If the file with the name is absent or not readable the program informs the user about it and goes to step4.
Actual Output	After the file was not found the program informed the user about it and went back to step 4.
Test Result	Successful

Table 11: Test 11

```
Choose three band members represented by codes separated by spaces:
JL AY DH
1) JL
2) AY
3) DH
Enter a number as per list for choosing a band member you want:3
Error! Can not find file!

SELECT A BAND
```

Band	code
Beatles	BEA
AC/DC	AD
Queen	QUE
Blondie	BLO
Nirvana	NIR

```
Which is the best music band:-BEA
```

Figure 11: To pick a member with non-existing file

**Test 12:** select existing external file

Test No.	12
Objective	To select existing external file
Input	Script was opened <ul style="list-style-type: none"> <li>Existing file number was entered</li> </ul> Argument 3
Expected Output	The program displays the code with a comment on the same line and then displays 3-4 lines description of that player.
Actual Output	The program displays the code with comment and gives 3-4 lines description of the player.
Test Result	Successful

Table 12: Test 12

```

hyper@DESKTOP-L887D84: ~
The Five Star band members are:

| Members          | code |
|-----|-----|
| John Lennon      | JL   |
| Angus Young      | AY   |
| Freddie Mercury | FM   |
| Debbie Harry     | DH   |
| Kurt Cobain       | KC   |

Choose three band members represented by codes separated by spaces:
JL AY KC
1) JL
2) AY
3) KC
Enter a number as per list for choosing a band member you want:3

You selected KC

Kurt Donald Cobain was a singer, songwriter, and artist from the United States. Nirvana's guitarist,
lead vocalist, and primary songwriter was Kurt Cobain. Cobain's works expanded the conceptual patterns
of popular rock through his angst-fueled lyricism and anti-establishment image. He was hailed as a voice
for Generation X and is regarded as one of the most important musicians in alternative rock history.
Do you want to start over?(y/n):-n

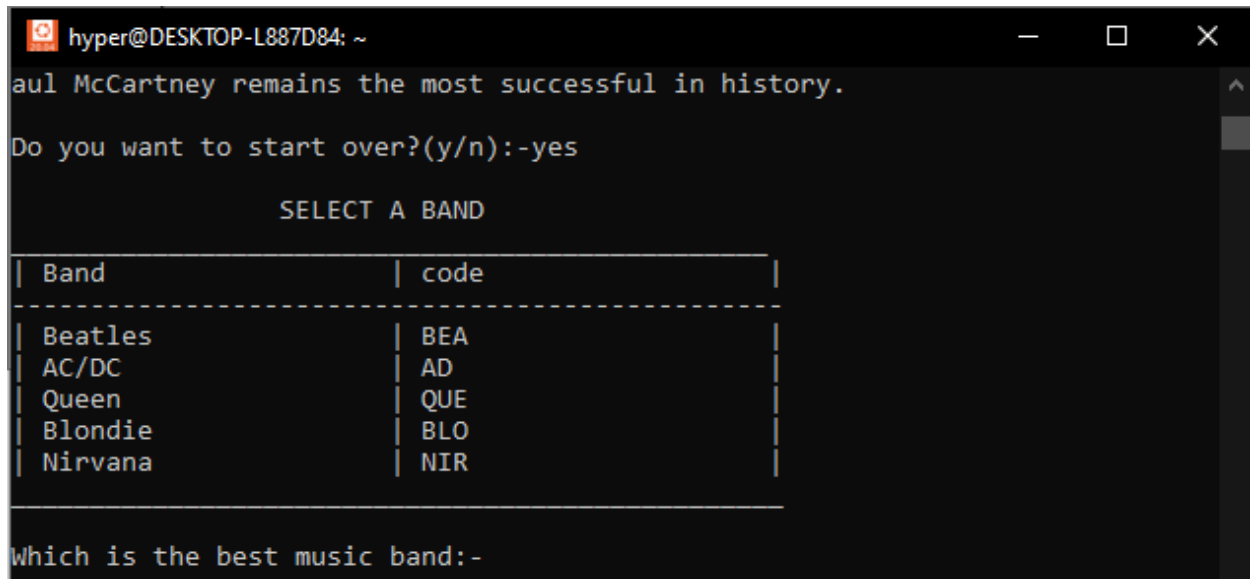
```

Figure 12: select existing external file

**Test 13:** Choosing yes when asked to start over

Test No.	13
Objective	Choose yes when asked to start over
Input	Script file was started. <ul style="list-style-type: none"> <li>Yes, was written when asked to repeat steps from 4 to 11</li> </ul> Argument yes
Expected Output	The program asks whether to repeat the 8 previous steps and if yes is selected it repeats it.
Actual Output	After yes was selected the program was repeated.
Test Result	Successful

Table 13: Test 13



```

hyper@DESKTOP-L887D84: ~
aul McCartney remains the most successful in history.
Do you want to start over?(y/n):-yes

      SELECT A BAND
+-----+-----+
| Band           | code |
+-----+-----+
| Beatles        | BEA  |
| AC/DC          | AD   |
| Queen          | QUE  |
| Blondie        | BLO  |
| Nirvana        | NIR  |
+-----+-----+
Which is the best music band:-

```

Figure 13: select existing external file



**Test 14:** Selecting no when asked to start over program.

Test No.	14
Objective	To select no when asked to repeat steps.
Input	Script file was started. <ul style="list-style-type: none"><li>• NO was entered when asked to repeat steps.</li></ul> Argument NO
Expected Output	After selecting no the program terminates.
Actual Output	When the user selected no the program was terminated.
Test Result	Successful

Table 14: Test 14

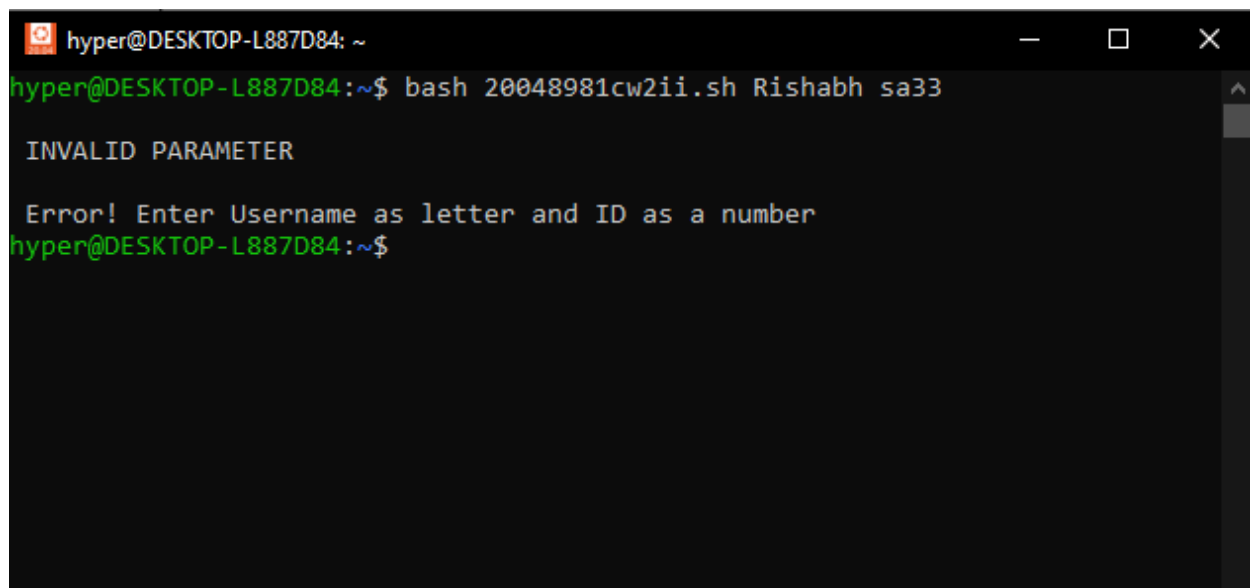
A screenshot of a terminal window titled 'hyper@DESKTOP-L887D84: ~'. The terminal shows a prompt 'Do you want to start over?(y/n):' followed by the input '-NO'. Below this, the program outputs 'Program Terminated Successfully' in a large, stylized, blocky font. The terminal prompt returns to 'hyper@DESKTOP-L887D84:~\$'.

Figure 14: Selecting no when asked to start over program.

**Test 15:** Entering wrong id (parameter validation)

Test No.	15
Objective	Entering wrong id (parameter validation)
Input	Script file was started <ul style="list-style-type: none"><li>Wrong id was entered.</li></ul> Argument Bash 20048981cw2ii.sh Rishabh sa33
Expected Output	After entering the wrong id the program notifies the user and does not execute.
Actual Output	When the user entered wrong id the program notified the user and did not execute.
Test Result	Successful

Table 15: Test 15



```
hyper@DESKTOP-L887D84: ~  
hyper@DESKTOP-L887D84:~$ bash 20048981cw2ii.sh Rishabh sa33  
  
INVALID PARAMETER  
  
Error! Enter Username as letter and ID as a number  
hyper@DESKTOP-L887D84:~$
```

Figure 15: Entering wrong id (parameter validation)

**Test 16:** Entering non existing number in member file selection

Test No.	16
Objective	Entering non existing number
Input	Script file was started <ul style="list-style-type: none"> <li>Wrong number was entered.</li> </ul> Argument 4,6,7,6,7,6
Expected Output	After entering the wrong number, the program notifies the user and the user will be able to type numbers.
Actual Output	When the user entered wrong number, the program notified the user and was able to type numbers.
Test Result	Successful

Table 16: Test 16

```

hyper@DESKTOP-L887D84: ~
The Five Star band members are:
| Members          | code |
|-----|-----|
| John Lennon      | JL   |
| Angus Young      | AY   |
| Freddie Mercury | FM   |
| Debbie Harry     | DH   |
| Kurt Cobain       | KC   |

Choose three band members represented by codes separated by spaces:
JL AY KC
1) JL
2) AY
3) KC
Enter a number as per list for choosing a band member you want:4
Please use correct number
Enter a number as per list for choosing a band member you want:6
Please use correct number
Enter a number as per list for choosing a band member you want:7
Please use correct number
Enter a number as per list for choosing a band member you want:6
Please use correct number
Enter a number as per list for choosing a band member you want:7
Please use correct number
Enter a number as per list for choosing a band member you want:6
Please use correct number
Enter a number as per list for choosing a band member you want:

```

Figure 16: Entering non existing number in member file selection

**Contents of three files: (TEXTS)**

- **JL**

John Lennon was an English singer, songwriter, musician and peace activist who attained worldwide renown as the founder, co-composer, co-lead vocalist and rhythm guitarist of the Beatles. Lennon was typified by the rebellious spirit and biting wit in his music, writing and drawings, on film, and in interviews. His songwriting partnership with Paul McCartney remains the most successful in history.

- **FM**

Freddie Mercury was a British singer-songwriter, who was best known as the main vocalist of the rock band Queen. He was regarded as one of the best singers in rock music history, famed for his flamboyant stage attitude and four-octave vocal range. Mercury's theatrical manner challenged rock singer traditions, shaping Queen's aesthetic direction.

- **KC**

Kurt Donald Cobain was a singer, songwriter, and artist from the United States. Nirvana's guitarist, lead vocalist, and primary songwriter was Kurt Cobain. Cobain's works expanded the conceptual patterns of popular rock through his angst-fueled lyricism and anti-establishment image. He was hailed as a voice for Generation X and is regarded as one of the most important musicians in alternative rock history.

## Conclusion

In conclusion the task A report is focused on creating a bash script which includes the requirements of the project. The bash script includes UNIX Script programming which checks the capability of coding program. It checks the ability to take inputs use conditional statement validate it etc. A testing was carried out and all the results were positive. Testing was done to check the practicality and usability of the script in real time scenario. This code taught me to use variables, conditional statements, loops, ask for user inputs, open files using script and many more. A deeper understanding was gained about bash scripting and its uses in real world scenario.

## Task B

### Introduction

Computer hardware and software are managed by an operating system (OS). Because computer hardware is so difficult to use and maintain, it necessitates extensive programming expertise. So, an operating system is designed to deal with hardware and make life easier for its users. To run an operating system, process management is a vital component. This includes duties such as creating and scheduling new processes, managing deadlock and terminating processes. The term "process" refers to a program that is currently in use. Child processes are those that are spawned from the parent process. A process needs access to system resources like CPU, memory and I/O devices to complete a given task. While a process is running it utilizes CPU and memory on a computer device. An operating system is responsible for running and managing all the process of the system. It manages operations by carrying out tasks such as process scheduling and resource allocation (Khan, 2008).

Process Management also protects resources of each process from other method and allows synchronization among process. A process has a collection of instructions that must be executed known as process code. The data to be processed is also linked to a process. Components of a process are the things that a process needs in order to function. The term "process state" refers to the state of a process at a specific point in time (Sukhla, 2017). Similar to these concepts, there are a lot of things to know about the process management function of an operating system, too. The following are a few examples of those concepts.

- Process State
- Process Control Block
- Process Scheduling
- Deadlock
- Process Operation

## Aims and Objectives

The aim of this coursework is to write a technical report that focuses on project management. A good culmination of good literature search work carried out using good mix of sources such as journals, industry white paper, web sources books etc.

The main objectives of the task b of this coursework are listed below.

- To gain knowledge of writing a technical report with proper format.
- Gather information form book, journals, industry white papers etc.
- Understand about different aspects of process management.
- To give proper referencing.
- Conducting proper research on Process Management.

## Background:

### Process Architecture

The hierarchical design of processes and systems used to transform inputs into outputs is referred to as process architecture. In simple terms while writing a computer program in a text file and when we execute the program it is called a process which performs all the tasks that is mentioned in the program. After a program is loaded on a memory it becomes a process. It can be divided into four sections as shown in the figure below.



Figure 17: Process architecture (Sites Bay, 2020)

- **Text Section:**

A text segment, often known as a code segment or simply as text, is a part of a program stored in an object file or memory that includes executable instructions. A text segment can be used as a memory space below the heap or stack to avoid heap or stack overflows from overwriting it. Text segments are typically sharable, requiring only a single copy to reside in memory for regularly executed applications such as text editors, the C compiler, shells, and so on. In addition, the text segment is frequently read-only to prevent a program from mistakenly changing its instructions.

- **Initialized Data Section:**

The initialized data segment contains initialized static variables, that is global variable and static variable. This segment is read write as the values can be altered at run time. As for the size of this segment it is determined by the size of the values which is situated in the source code of the program which does not change at run time.

- **Uninitialized Data Segment:**

The uninitialized data segment also known as the “bss” segment contains uninitialized static data and contains all global and static variables that are initialized to zero or do not have explicit initialization source code.

- **Heap:**

The heap segment is where the allocation of memory takes place, which may be processed during run time. Heap begins right after the end of uninitialized data segment “bss”.

- **Stack:**

The Stack contains temporary data including function parameters, returns addresses, and local variables. The Stack is used for local variables. Space on the stack is allotted for local variables when they are declared.



### Process Control Blocks

Process control block also known as task control block is a type of data structure that consists of information about the process related to it. Process control block also defines the current state of the OS. It is critical for process management since process data is structured in terms of the process control block. (PCB). The PCB is identified by an integer called Process ID. PCB helps to store information required to keep track of the running process. The important components of PCB are as follows (Onsman, 2018). The PCB is located in memory area which is protected from normal user access. The reason being that it contains important user information.

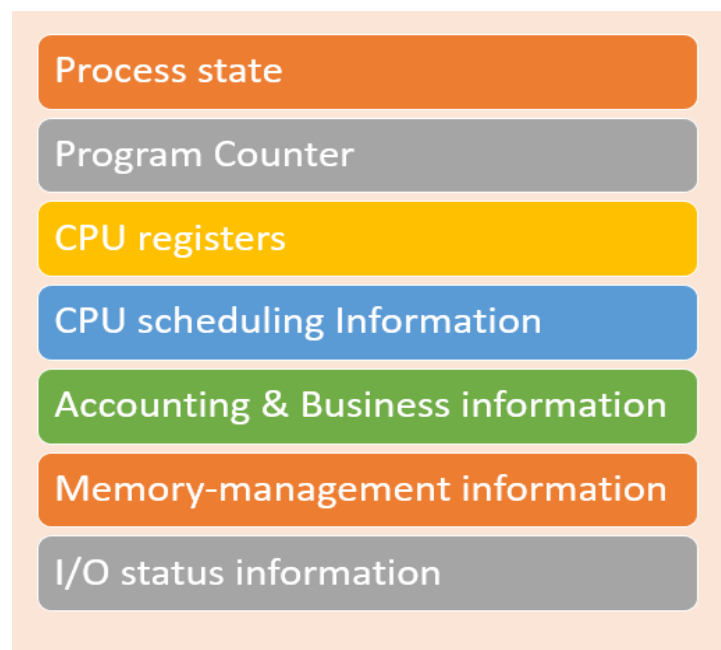


Figure 18: Process control block (PCB) (Williams, 2022)

- **Process State:**

The process state indicates the current state of the process which can be new, ready, running, terminating and waiting.

- **Program Counter:**

Program counter contains the address of the next instruction that needs to be executed for that process.

- **CPU registers:**

The number and kind of registers varies depending on the computer architecture. Accumulators, index registers, stack pointers, and general-purpose registers, as well as any condition code information, are all part of a register.

- **CPU scheduling information:**

This component of PCB includes the process priority, pointers for scheduling queues and other scheduling parameters.

- **Accounting and Business information:**

The accounting and business information includes the CPU usage, time limits, tasks, account holders or process numbers (Eres Publication, 2020).

- **Memory management information:**

The memory management information includes the value of information such as the value of base and limit registers, segment tables or the page. It depends on the memory system that is used by the O/S.

- **I/O status information:**

I/O status information is a block which contains the information on the process's listed input output devices, a list of open files etc.

**Process States**

A process it goes through various states from its creation to completion. It defines the current position of a process. The process state helps us to get the detail of the process at a particular instant. There are seven states of process which are listed below.

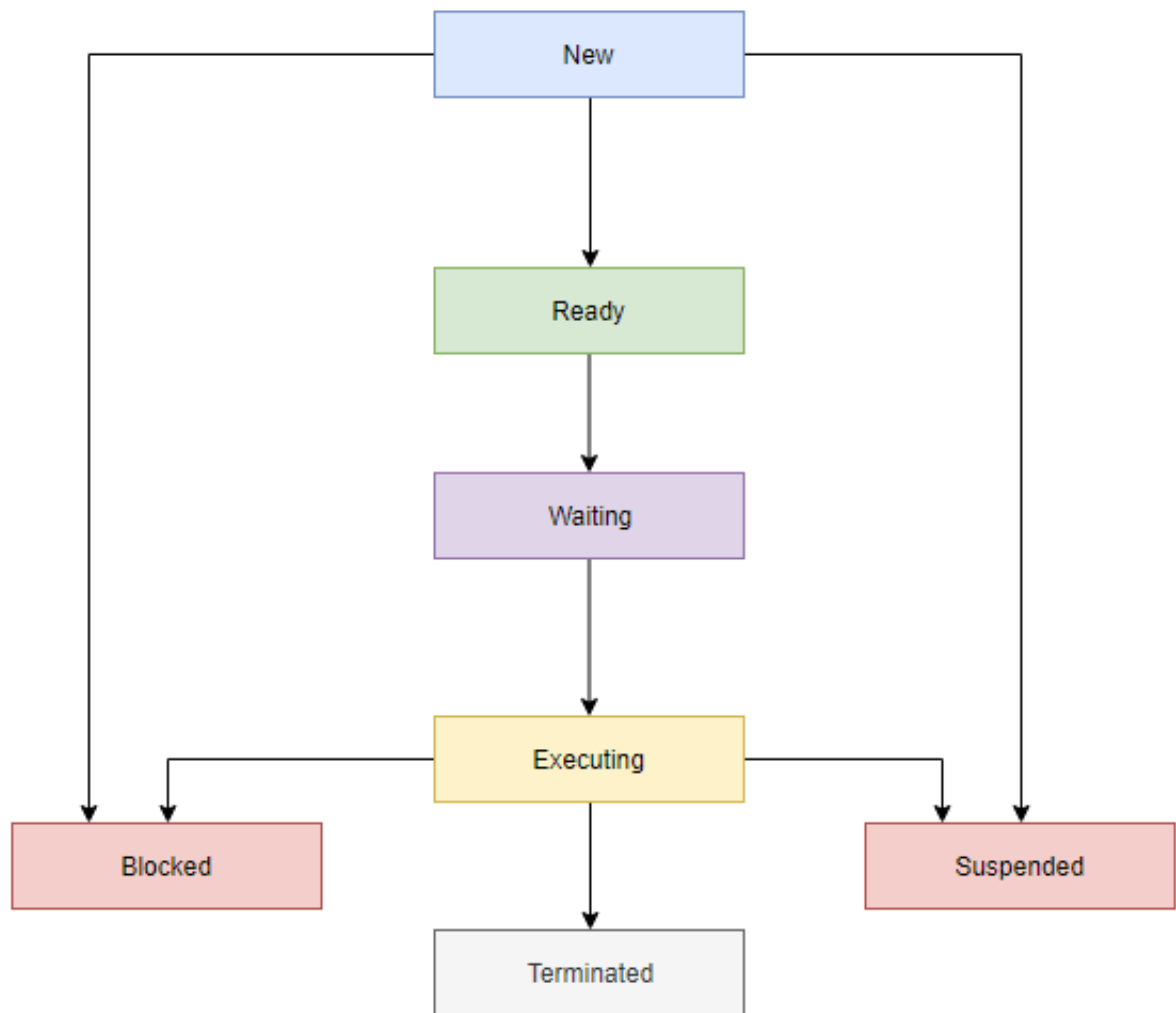


Figure 19: Process States (Williams, 2022)

- **New:**

The state when a process is about to be created but not yet created. New process is created when a program calls from secondary memory to primary memory. It is the first state of the process.

- **Ready:**

When the process is created it goes to ready state then process is loaded into the primary memory and is ready for execution. Processes that are ready for CPU execution are kept in a queue for ready processes. There can be more than one process in a ready state.

- **Waiting:**

In the waiting state the process is waiting for allocation of the CPU time and other resources for its execution. It simply means that that the process is in hold and the other process are allowed to start their execution.

- **Executing:**

Executing state is the state when a process is being executed by the CPU. It is a main state of any process. The process is picked by the CPU using some CPU scheduling algorithm.

- **Blocked:**

In this time interval the process is waiting for an event like input/output operations (I/O) or some more priority process might come that needs to get completed. In this case other process will come for execution and the running process will have to go in waiting.

- **Suspended:**

Processes that were initially in the ready state but were switched out of main memory by the scheduler and placed on external storage are said to be in the suspend state (Dusey, 2022).

- **Terminated:**

The termination state is when the process is ended or terminated and the resources or the memory that are utilized by the process are free (Sukla, 2017). In simple terms after the complete execution of the process it comes to a terminated state and all the information related to it is deleted.

### Process Hierarchies

The modern-day operating system (O/S) creates many processes and destroy many processes. A process creates different process during their execution. When one process generates another process, the parent and child processes tend to associate with each other in specific ways. The child process can also create other processes if necessary. This parent-child structure of processes forms a hierarchical structure known as Process Hierarchy. A process and all of its children and grandchildren are referred to as a process group in UNIX. When a user transmits a signal via the keyboard, it is sent to all members of the process group currently associated with the keyboard. Each process can catch the signal and perform whatever it wants with it (Pal, 2020). As of windows there is no hierarchy system each and every process in windows is created equally. In UNIX, this is accomplished through the use of the Fork system call, which starts a child process, and the exit system call, which ends the current process. The root of the tree is a special process that the operating system creates during start up.

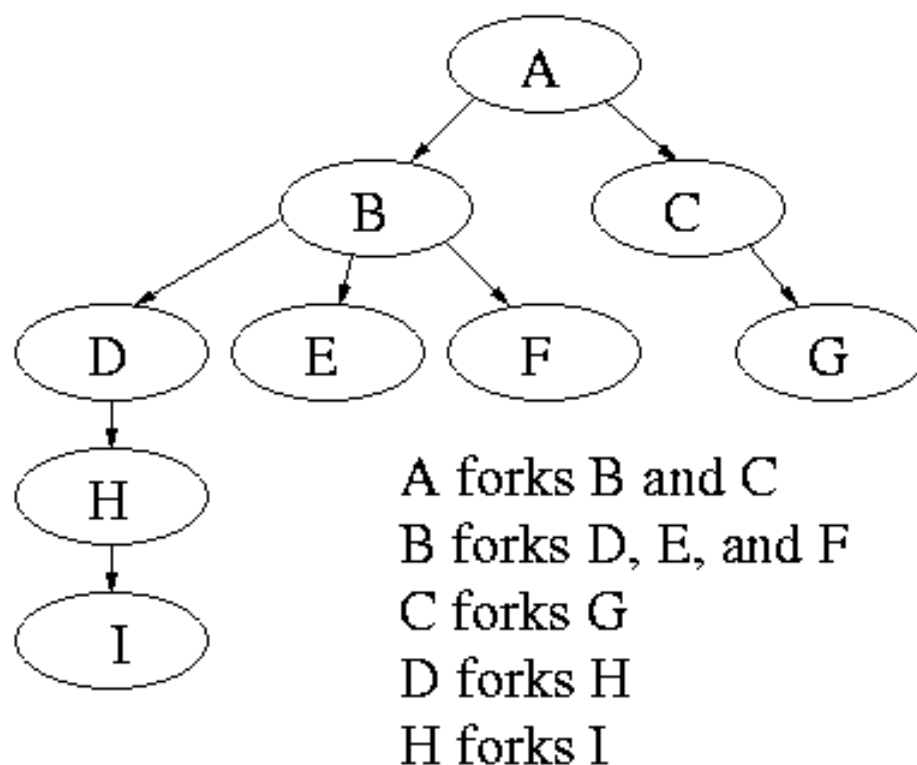


Figure 20: Process Hierarchies (Anon., 2002)

## Implementation of process

In order to implement a process, the operating system keeps a table called the process table, which provides head to toe information on the process. The process table is used by the operating system (O/S) to keep track of the processes in the system. The table entry below contains information about the processes' states, program counters, stack pointers, allocated memories, accounting and scheduling information, and file status information. They contain information about both the present processes and those that are swapped with the current processes (Pal, 2020). Below are some of the fields of a process table.

<b>Process management</b>	<b>Memory management</b>	<b>File management</b>
Registers	Pointer to text segment info	Root directory
Program counter	Pointer to data segment info	Working directory
Program status word	Pointer to stack segment info	File descriptors
Stack pointer		User ID
Process state		Group ID
Priority		
Scheduling parameters		
Process ID		
Parent process		
Process group		
Signals		
Time when process started		
CPU time used		
Children's CPU time		
Time of next alarm		

Figure 21: Process table (Athiralekshmicv, 2014)

As for the table the fields in the first column are related to process management. The second column is related to memory management and finally the last column is related to file management.

## Conclusion

In conclusion process management is one of the core modules of an O/S. The process management deals with jobs like creating and scheduling new processes, managing deadlock and terminating processes. A process is a program that is currently in use. The process utilizes computer resources while it is running. The operating system is responsible for managing the process. The process architecture means the hierarchical design of processes and systems used to transform inputs into outputs. When you are writing a computer program in a text file and when we execute the program it is called a process which performs all the tasks that is mentioned in the program. Let's move on to the next topic Process control block (PCB). It is critical for process management since process data is structured in terms of the process control block. Process scheduling goes thorough different stages from its creation to execution. It helps to determine the current state of the operating system.

Moving further on we have process hierarchy its when a process created another process. It is very useful in modern day operating systems because the O/S creates a lot of process and deleted a lot of process. Process hierarchy is seen in UNIX, this is accomplished through the use of the Fork system call, which starts a child process, and the exit system call, which ends the current process. But in windows we don't see the concept of hierarchy all the process are created equally. Now for the implementing of the process, the operating system maintains a table called process table which contains head to toe information of a process in order to implement a process. Proper research was carried out for the completion the report. The table is used by the operating system is used to keep track of the processes in the system. A deep knowledge of process management and its aspects was gained.



## References

Anon., 2002. *NYU*. [Online]

Available at: <https://cs.nyu.edu/~gottlieb/courses/2000s/2001-02-spring/os/lectures/lecture-05.html>

[Accessed 25 April 2022].

Anon., n.d. *Tallyfy*. [Online]

Available at: <https://tallyfy.com/process-architecture/>

[Accessed 24 April 2022].

Athiralekshmicv, 2014. *myaspirationsanddreams*. [Online]

Available at: <https://myaspirationsanddreams.wordpress.com/2014/06/05/processes-3/>

[Accessed 25 April 2022].

Dang Van Duc, V. D. L., 2005. *Operating Syatems*. s.l.:Institute of information and technology.

Data Flair, 2022. *Data Flair*. [Online]

Available at: <https://data-flair.training/blogs/priority-scheduling-algorithm-in-operating-system/>

[Accessed 25 April 2022].

Dusey, A., 2022. *geeks for geeks*. [Online]

Available at: <https://www.geeksforgeeks.org/states-of-a-process-in-operating-systems/>

[Accessed 24 April 2022].

Eres Publication, 2020. *erespublications*. [Online]

Available at: <https://erespublications.blogspot.com/2020/07/explain-process-control-block-with.html>

[Accessed 25 April 2022].

Free Software Foundation, Inc., 2008. *gnu*. [Online]

Available at: [https://www.gnu.org/software/bash/manual/html\\_node/GNU-Free-Documentation-License.html](https://www.gnu.org/software/bash/manual/html_node/GNU-Free-Documentation-License.html)

[Accessed 25 April 2022].

Hailperin, M., 2007. *Operating Syatems*. 5th ed. San Francisco: Thomas Course Technology.

Hira, Z., 2022. *free code camp*. [Online]

Available at: <https://www.freecodecamp.org/news/shell-scripting-crash-course-how-to-write-bash-scripts-in-linux/>

[Accessed 25 April 2022].

Javatpoint, 2022. *javatpoint*. [Online]

Available at: <https://www.javatpoint.com/os-priority-scheduling>

[Accessed 25 April 2022].

Khan, Y. I., 2008. Process Management. *Formal Modeling of Process Management System*, p. 7.

Mahmood, T., n.d. *Operating Syatems and Networks*. 5th ed. Faisalabad: IT series.

Onsman, A., 2018. *tutorialspoint*. [Online]

Available at: <https://www.tutorialspoint.com/what-is-process-control-block-pcb>  
[Accessed 24 April 2022].

Onsman, A., 2018. *tutorialspoint*. [Online]

Available at: <https://www.tutorialspoint.com/what-is-process-control-block-pcb>  
[Accessed 25 April 2022].

Pal, T., 2020. *technobyte*. [Online]

Available at: <https://technobyte.org/processes-in-operating-systems/>  
[Accessed 25 April 2022].

Sites Bay, 2020. *Sitesbay*. [Online]

Available at: <https://www.sitesbay.com/os/os-process-states-in-operating-system>  
[Accessed 25 April 2022].

Sukhla, A., 2017. *includehelp*. [Online]

Available at: <https://www.includehelp.com/operating-systems/process-management-of-operating-system.aspx>  
[Accessed 23 April 2022].

Sukla, A., 2017. *includehelp*. [Online]

Available at: <https://www.includehelp.com/operating-systems/process-management-of-operating-system.aspx>  
[Accessed 24 April 2022].

Williams, L., 2022. *Guru99*. [Online]

Available at: <https://www.guru99.com/process-management-pcb.html>  
[Accessed 25 April 2022].

## Appendix

### Appendix – A (Glossary)

### Appendix – B (Process Scheduling)

Process Scheduling is the process of removing a current running task from the processor and adding another task to process. It helps to divide a process into different states such as running, waiting and ready. A process goes through many steps from its creation to completion. A separate queue is made with the PCB by the operating system. Process control block is moved to a new state queue as it is unlinked from the current queue after the state of process changes. Some of the process scheduling queues are as follows.

- Ready queue: It is ready and waiting for execution. A set of all processes is stored in main memory any new process that comes is stored here.
- Job queue: This helps to store all the process available in the system.
- Device queue: The device queue consists of blocked process due to the absence of input output devices.

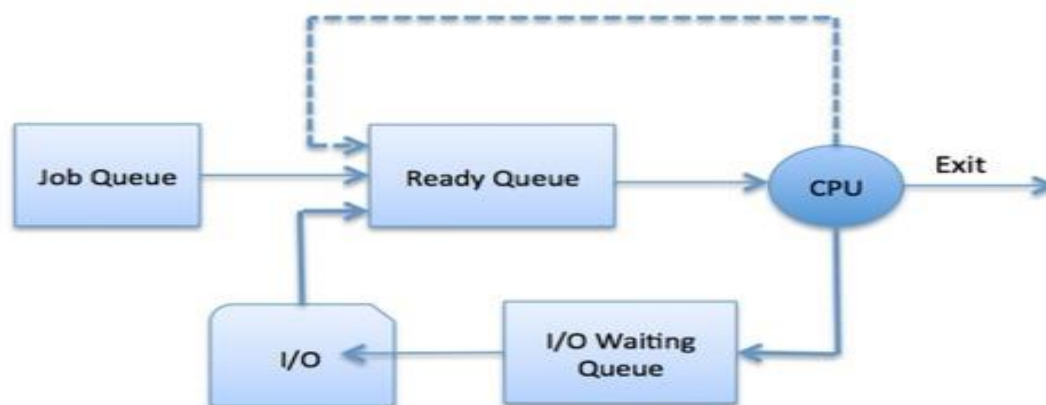


Figure 22: Process Scheduling

Some of the objectives of process scheduling are it does not postpone for a undetermined time and enforces priorities. It increases the number of interactive users while maintaining acceptable response times. It achieves a good balance of reaction

and utilization. It refers to the procedures that is holding key resources (Data Flair, 2022).

### Appendix – c (Priority Scheduling)

Priority scheduling is a process of scheduling a process using the process scheduling algorithm which is based on priority the scheduler then selects the tasks according to its priority. Every process gets a number to know its priority level. If there are two jobs that have to be executed and has the same priorities, then the process is chosen on the basis of first come first serve (FCFS) or the round robin. The priority of a process depends on many factors like the time requirement, memory requirement etc. There are two types of process scheduling they are as follows (Data Flair, 2022).

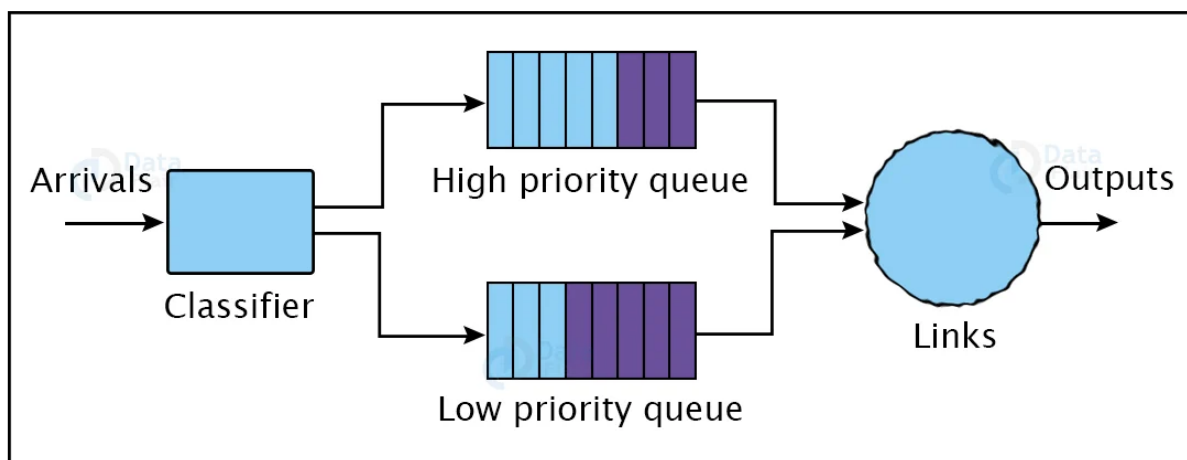


Figure 23: Priority Scheduling (Data Flair, 2022)

- Pre-emptive Scheduling:

In pre-emptive scheduling if a higher priority task arrives while a lower priority task is running the lower priority task will be put on hold. After the high priority task is completed the lower priority task resumes from the place it was paused. This scheduling requires a special kind of hardware called timer.

- Non-Pre-emptive Scheduling

In this scheduling the process are schedules according to their priority numbers. After the process is schedules, it will run till the completion. In this case if the priority number is lower then the process priority gets higher.

Some of the benefits of priority scheduling are its easy to use. The process importance is clearly stated. It saves time and is suitable for application whose time and resource requirement differ frequently.

Some of the disadvantages are the lower process hold time can be indefinite, the high priority resources use a lot of resources, if the system crashes we can lose the low priority processes.