Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Short Answer Question

What is the recall of class 0? (in percentage)

Hint: Round your answer to two decimal places and answer them in terms of percentage. Example: If your answers is 0.735, write it as 73.50.

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Range

Text Areas : PlainText

Possible Answers :

39.9 to 40.1

System Commands

Section Id :	64065356699
Section Number :	14
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	16
Number of Questions to be attempted :	16
Section Marks :	100
Display Number Panel :	Yes
Section Negative Marks :	0
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	Yes
Maximum Instruction Time :	0

Sub-Section Number : 1

Sub-Section Id : 640653118970

Question Shuffling Allowed : No

Is Section Default? : null

Question Number : 392 Question Id : 640653816273 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "DIPLOMA LEVEL : SYSTEM COMMANDS (COMPUTER BASED EXAM)"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?
CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS REGISTERED BY YOU)

Options :

6406532734257. ✔ YES

6406532734258. ✖ NO

Sub-Section Number : 2

Sub-Section Id : 640653118971

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 393 Question Id : 640653816274 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

Select the script to find and kill the process which is consuming more than 80% of the CPU.

Hint:

```
$ ps -eo pid,%cpu --sort=-%cpu
    PID %CPU
 243099 88.2
   9640  7.0
   9814  3.1
   2824  2.8
 242251  2.4
   9822  1.2
   3140  0.7
   2654  0.7
   9902  0.6
```

- `xargs` is used to pass the stdout of one command as arguments to another command.

Options :

6406532734259. ✔ `ps -eo pid,%cpu --sort=-%cpu | awk '$2 > 80 {print $1}' | xargs kill`

6406532734260. ✖ `ps -eo pid,%cpu --sort=-%cpu | awk '$2 < 80 {print $1}' | xargs kill`

6406532734261. ✖ `ps -eo pid,%cpu --sort=-%cpu | awk '$0 > 80 {print $1}' | xargs kill`

6406532734262. ✖ `ps -eo pid,%cpu --sort=-%cpu | awk '$0 < 80 {print $1}' | xargs kill`

Question Number : 394 Question Id : 640653816275 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

What does the following script do?

```
while read -r line; do
    [ "$line" = EOF ] && break
    echo $line
done <file1 >file2
```

Options :

6406532734263. ✖ Copies the contents of `file1` to `file2`.

6406532734264. ✔ Copies the contents of `file1` to `file2` until the line `EOF` is encountered and the line `EOF` is not copied.

6406532734265. ✖ Copies the contents of `file1` to `file2` until the line `EOF` is encountered, the line `EOF` is copied.

6406532734266. ✖ Copies the contents of `file1` to `file2` until the last occurrence of the line `EOF` is encountered and the line `EOF` is not copied.

Question Number : 395 Question Id : 640653816278 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

What does the following script do?

```
#!/bin/bash

x=0
for number in {10..20}; do
    # The ends of the range are inclusive
    if ((number % 2 != 0)); then
        x=$((x + number))
    fi
done
echo $x
```

Options :

6406532734269. ✖ Prints the count of even numbers between 10 and 20.

6406532734270. ✖ Prints the count of odd numbers between 10 and 20.

6406532734271. ✖ Prints the sum of even numbers between 10 and 20.

6406532734272. ✔ Prints the sum of odd numbers between 10 and 20.

Question Number : 396 Question Id : 640653816279 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

The PATH environment variable:

- Stores a list of directories separated by colons (":").
- The shell searches for executable commands in these directories, one by one, in the order they appear in the variable PATH.

Select the command that does the following:

Here, the script `my-script` is taken as an **example**. If the command `my-script` is run:

1. Check for `my-script` in `~/.local/bin`:
   - If an executable file named `my-script` exists in the `~/.local/bin` directory, run that specific file.
2. Check for `my-script` in `/opt/extra/bin` (if not found in step 1):
   - If the executable file `my-script` doesn't exist in `~/.local/bin` but exists in `/opt/extra/bin`, run the one from `/opt/extra/bin`.
3. The same flow should be followed if the command is run from a script.

Options :

6406532734273. ✖
```
export PATH=/opt/extra/bin:~/.local/bin:$PATH
```

6406532734274. ✔
```
export PATH=~/.local/bin:/opt/extra/bin:$PATH
```

6406532734275. ✖
```
PATH=/opt/extra/bin:~/.local/bin:$PATH
```

6406532734276. ✖ `PATH=~/.local/bin:/opt/extra/bin:$PATH`

Question Number : 397 Question Id : 640653816280 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

Consider the following shell script:

```
# myscript.sh
for file in *.txt; do
    echo "Processing $file"
    cp "$file" "backup_$file"
done
```

What does this script do?

Options :

6406532734277. ✖ Renames all `.txt` files by adding the prefix “backup_” to their names.

6406532734278. ✖ Deletes all `.txt` files.

6406532734279. ✔ Creates backup copies of all `.txt` files.

6406532734280. ✖ Moves all `.txt` files to a backup directory.

Question Number : 398 Question Id : 640653816281 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

What environment variable stores the path to the directory containing the commands?

Options :

6406532734281. ✖ `$HOME`

6406532734282. ✔ `$PATH`

6406532734283. ✖ `$USER`

6406532734284. ✖ `$PWD`

Question Number : 399 Question Id : 640653816282 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

In a text file, you want to replace the **second occurrence** of the word "apple" with "banana". Which `sed` command would you use?

Options :

6406532734285. ✔ `sed 's/apple/banana/2' filename`

6406532734286. ✖ `sed 's/apple/banana/g2' filename`

6406532734287. ✖ `sed '2s/apple/banana/' filename`

6406532734288. ✖ `sed '2s/apple/banana/g' filename`

Sub-Section Number : 3

Sub-Section Id : 640653118972

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 400 Question Id : 640653816284 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6 Max. Selectable Options : 0

Question Label : Multiple Select Question

You want to find lines in a text file where the third field is a numeric value between 57 and 93 inclusive. Which command(s) would you use?

Options :

6406532734293. ✔ `awk '$3 >= 57 && $3 <= 93 {print $0}' filename`

6406532734294. ✖ `awk '$3 ~ /^[57-93]+$/ {print}' filename`

6406532734295. ✖ `awk '$3 ~ /^[0-9]{2}$/ {print}' filename`

6406532734296. ✔ `awk '$3 >= 57 && $3 <= 93' filename`

6406532734297. ✔ `awk '$3 ~ /[[:digit:]]+/ && $3 >= 57 && $3 <= 93 {print}' filename`

Question Number : 401 Question Id : 640653816285 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6 Max. Selectable Options : 0

Question Label : Multiple Select Question

Select the correct option(s) to validate whether a string variable contains only digits, dot and

minus in a bash script.

Options :

6406532734298. ✔ `[[ $variable =~ ^[0-9.\-]+$ ]]`

6406532734299. ✖ `[[ $variable =~ ^[0-9.\-]*$ ]]`

6406532734300. ✔ `[[ $variable =~ ^[[:digit:].\-]+$ ]]`

6406532734301. ✖ `[[ $variable =~ ^[[:digit:].\-]* ]]`

Sub-Section Number :	4
Sub-Section Id :	640653118973
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 402 Question Id : 640653816283 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 8 Max. Selectable Options : 0

Question Label : Multiple Select Question

You need to extract all lines which contain URLs from an HTML document. Which regular expression (ERE) would you use?

Hint

Use the sample input and output for your reference.

sample input

```
<!doctype html>
<html>
  <head>
    <title>Sample HTML Document</title>
  </head>
  <body>
    <h1>Welcome to my website!</h1>
    <p>Here are some links:</p>
    <ul>
      <li><a href="https://www.example.com">Example Website</a></li>
      <li><a href="http://www.test.com/page1">Test Page 1</a></li>
      <li><a href="https://www.test.com/page2">Test Page 2</a></li>
      <li><a href="ftp://ftp.example.com">FTP Server</a></li>
      <li><a href="mailto:info@example.com">Contact Us</a></li>
    </ul>
  </body>
</html>
```

Sample output

```
<li><a href="https://www.example.com">Example Website</a></li>
<li><a href="http://www.test.com/page1">Test Page 1</a></li>
<li><a href="https://www.test.com/page2">Test Page 2</a></li>
<li><a href="ftp://ftp.example.com">FTP Server</a></li>
<li><a href="mailto:info@example.com">Contact Us</a></li>
```

Options :

6406532734289. ✖ `<a href="([^"]+)">`

6406532734290. ✔ `<a href="([^"]*?)">`

6406532734291. ✖ `<a href="([^"]*)">`

6406532734292. ✔ `<a href="([^"]+?)">`

Question Number : 403 Question Id : 640653816286 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 8 Max. Selectable Options : 0

Question Label : Multiple Select Question

Arrange the following steps in the correct order to create an AWK script that calculates the mean, median, and mode of a given dataset stored in a text file named "data.txt":

1. Use awk to read the data from the text file and store it in an array.
2. Calculate the mean by summing up all the values in the array and dividing by the total number of values.
3. Sort the array to find the median value.
4. Calculate the mode by counting the frequency of each unique value in the array.
5. Determine the value with the highest frequency as the mode.
6. Print the mean, median, and mode to the terminal.
7. Prompt the user to enter the filename containing the dataset.
8. Check if the specified file exists and is accessible.

Options :

6406532734302. ✔ 7 -> 8 -> 1 -> 2 -> 3 -> 4 -> 5 -> 6

6406532734303. ✖ 7 -> 8 -> 1 -> 4 -> 2 -> 3 -> 5 -> 6

6406532734304. ✖ 8 -> 7 -> 1 -> 2 -> 3 -> 4 -> 5 -> 6

6406532734305. ✖ 8 -> 7 -> 1 -> 4 -> 2 -> 3 -> 5 -> 6

6406532734306. ✔ 7 -> 8 -> 1 -> 3 -> 2 -> 4 -> 5 -> 6

Question Number : 404 Question Id : 640653816287 Question Type : MSQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 8 Max. Selectable Options : 0

Question Label : Multiple Select Question

Choose the corner case(s) from stdin that makes this SED command fail to replace all the three-character month names with the corresponding numbers.

```
#!/bin/bash

# Associative array
declare -A month_to_number

month_to_number=(
        ["Jan"]=1 ["Feb"]=2 ["Mar"]=3 ["Apr"]=4
        ["May"]=5 ["Jun"]=6 ["Jul"]=7 ["Aug"]=8
        ["Sep"]=9 ["Oct"]=10 ["Nov"]=11 ["Dec"]=12
) # ([key]=value)

read -r line
for m in "${!month_to_number[@]}"; do
        # get from stdin
        [[ "$line" =~ $m ]] || continue
        echo "$line" | sed "s/$m/${month_to_number[$m]}/"
done
```

Options :

6406532734307. ✖ `12/Jan/2017`

6406532734308. ✖ `22-Aug-1999 Morning`

6406532734309. ✔ `17/Feb/1888, 7/May/1999`

6406532734310. ✔ `18/jul/2047`

6406532734311. ✖ `19 Sep 2023 18:00`

Sub-Section Number : 5

Sub-Section Id : 640653118974

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 405 Question Id : 640653816276 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 7

Question Label : Short Answer Question

How long will the following script run?

Hint:

- `$!` expands to the PID of the most recently executed **background command**.
- Provide the answer in integer format.

```
for i in {6..15}; do
    sleep 3 &
    sleep 1 && kill $!
done
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

10

Question Number : 406 Question Id : 640653816277 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 7

Question Label : Short Answer Question

What will be the output of the last command? [NAT]

Note: The answer is case-sensitive, so provide it without leading or trailing spaces.

```
$ cat data.txt
Hello World
$ cat data.txt | while read -r line; do
    echo "${line% *}"
done
```

Response Type : Alphanumeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Answers Case Sensitive : No

Text Areas : PlainText

Possible Answers :

Hello

Sub-Section Number :	6
Sub-Section Id :	640653118975
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 407 Question Id : 640653816288 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 8

Question Label : Short Answer Question

What will be the output from the following script?

Note: Provide the answer in integer format.

```
awk '
{
    x[NR]=$1; y[NR]=$2
    x_+=$1;   y_+=$2
}
END {
    x_=x_/NR; y_=y_/NR
    denx_2=0; deny_2=0
    for (i in x) {
        num+=(x[i]-x_)*(y[i]-y_)
        denx_2+=(x[i]-x_)^2
        deny_2+=(y[i]-y_)^2
    }
    print num/(denx_2^0.5 * deny_2^0.5)
}
' << EOF
1 2
3 4
5 6
7 8
EOF
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

1

TDS

Section Id :	64065356700
Section Number :	15
Section type :	Online
Mandatory or Optional :	Mandatory