

3. Download the file screenshots.tar.gz from course page on the moodle site. The screenshots contain images of the terminal window with few commands that expose the hardware / configuration details of my laptop I use for this course instruction. You need to make a clean pdf that contains two terminal images per page. This involves trimming of the images. Your script should take the pixel values for trimming as a user input and then do the task automatically for all the images in the directory and generate the pdf.

The solution includes the script and one sample page the output pdf. [2 Marks]

Hint: Install imagemagick with `sudo apt install imagemagick` and check man page of `convert` command.

Application: You may need to trim a large set of images from an experiment and use them for image analysis, to create a video sequence and to extract events. High speed imaging of experiments is an important scientific technique.

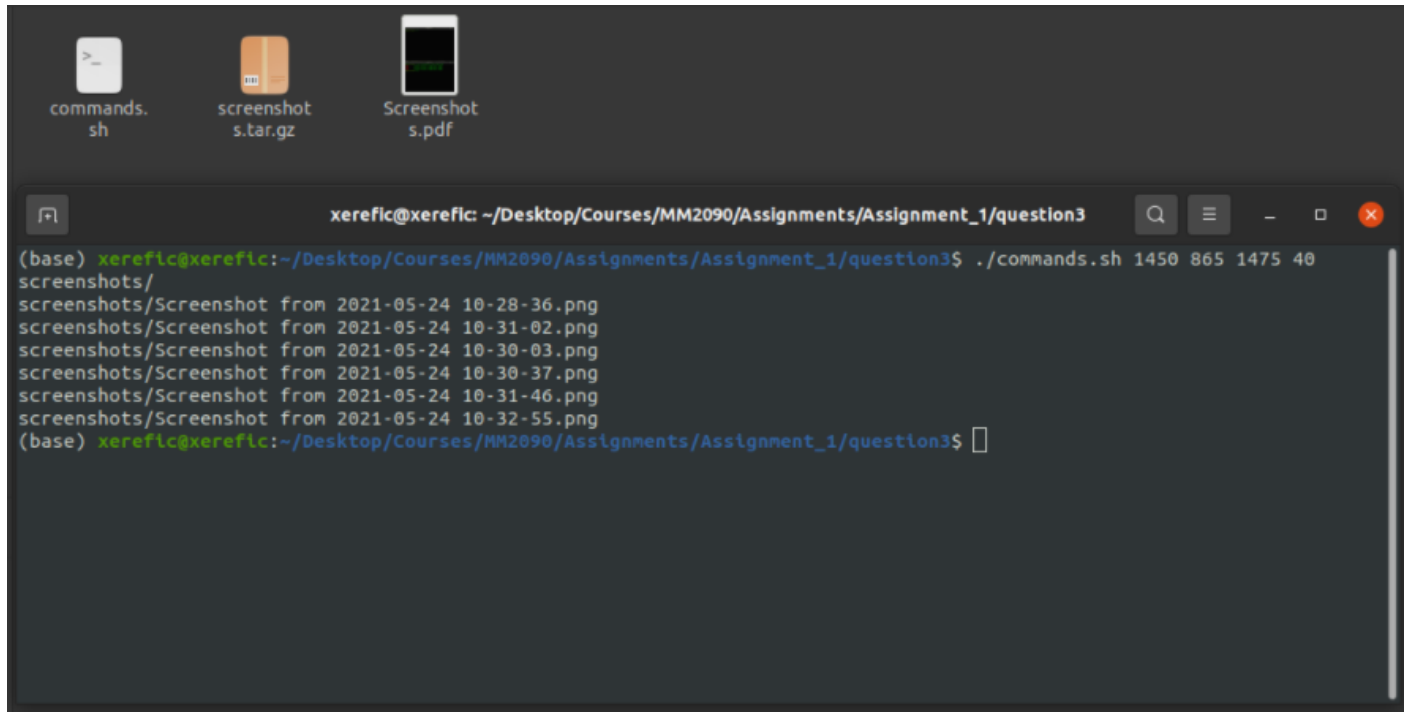
Link to the GitHub repository for this question: [GitHub](#)

The bash script takes parameters – width, height and strides and outputs the pdf of the cropped images.

```
1. #!/bin/bash
2.
3. # Unzipping the screenshots
4. tar -xvzf screenshots.tar.gz
5. mkdir cropped
6.
7. # Renaming the files for ease of access
8. cd screenshots/
9. for f in *\ *; do mv "$f" "${f// /_}"; done
10. i=0;
11. for files in `ls`;
12. do
13.     mv $files $(basename $i).png;
14.     i=$(( $i+1 ));
15. done;
16.
17. # size: 1450x865+1475+40
18. cd ..
19.
20. # Cropping the images
21. for files in `ls screenshots/`;
22. do
23.     if [ -z "$1" ]; then
24.         width=`identify -ping -format '%w' screenshots/$files`
25.         height=`identify -ping -format '%h' screenshots/$files`
26.         x=0
27.         y=0
28.     else
29.         width=$1
30.         height=$2
31.         x=$3
32.         y=$4
33.     fi
34.     convert screenshots/$files -crop $width\x$height+$x+$y cropped/$(basename $files .png).png
35. done;
36.
37. rm -r screenshots/
38. mkdir combined
39.
40. # Combining two images into one by appending them vertically
41. for j in $(seq 0 $((($i/2-1)));
42. do
43.     convert -append cropped/$(basename $((($j*2))).png) cropped/$(basename $((($j*2+1))).png
44.         combined/$(basename $j).png
45.     done;
46.
47. rm -r cropped/
48.
49. # Joining the combined images into a pdf
50. convert -page A4 -resize 3508x2480 combined/*.png Screenshots.pdf
51.
```

```
52. rm -r combined/  
53.
```

TERMINAL:



The image shows a terminal window on a desktop environment. The desktop has three icons: 'commands.sh', 'screenshot.s.tar.gz', and 'Screenshot.s.pdf'. The terminal window title is 'xerefic@xerefic: ~/Desktop/Courses/MM2090/Assignments/Assignment\_1/question3'. The terminal output shows the execution of './commands.sh' which lists several PNG files in the 'screenshots/' directory, all dated 2021-05-24. The prompt then returns to the shell.

```
(base) xerefic@xerefic:~/Desktop/Courses/MM2090/Assignments/Assignment_1/question3$ ./commands.sh 1450 865 1475 40  
screenshots/  
screenshots/Screenshot from 2021-05-24 10-28-36.png  
screenshots/Screenshot from 2021-05-24 10-31-02.png  
screenshots/Screenshot from 2021-05-24 10-30-03.png  
screenshots/Screenshot from 2021-05-24 10-30-37.png  
screenshots/Screenshot from 2021-05-24 10-31-46.png  
screenshots/Screenshot from 2021-05-24 10-32-55.png  
(base) xerefic@xerefic:~/Desktop/Courses/MM2090/Assignments/Assignment_1/question3$
```

OUTPUT: