CIS*2520 Lab 2 - Complexity (F2024, Week of Sept. 23-27)

1. Suppose you have a computer that requires 1 minute to solve problem instances of size n = 100. What instance sizes can be run in 1 minute if you buy a new computer that runs 64 times faster than the old one, assuming the Time complexities T (n) ∈ _$\boldsymbol{\Theta}$ _(2^n) for the algorithm?

Solution:

$$\frac{c * (2)^{100}}{c * (2)^{x}} = \frac{1}{64}$$

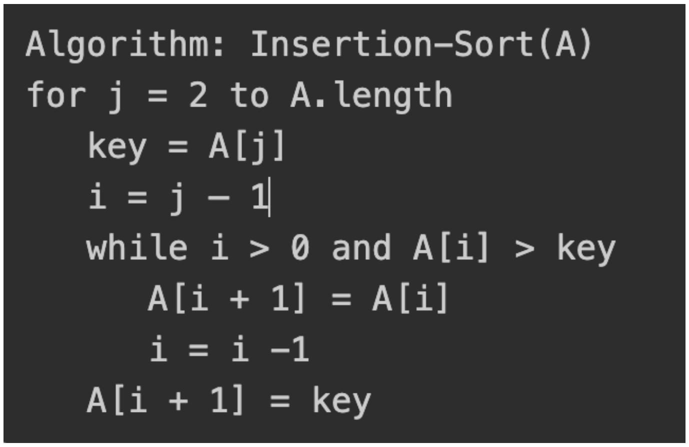
$$(2)^{x} = (2)^{100} * 64 = (2)^{100} * (2)^{6}$$

$$x = 106$$

2. Calculate the best case and worst case total operations for the following insertion-sort algorithm for an integer array A with length n.

Note: each line of the pseudocode can be counted as 1. For example, each of the following is counted as 1:

- **a) i=1;i<=n;i++ (not 3)**
- **b) x=b**
- **c) i>0 and j<0 (not 2)**



```
Algorithm: Insertion-Sort(A)
for j = 2 to A.length
   key = A[j]
   i = j − 1
   while i > 0 and A[i] > key
      A[i + 1] = A[i]
      i = i −1
   A[i + 1] = key
```

Solution:

For simplicity, the best case input can be represented as $[1,2,3,\ldots,n]$, and the worst can be $[n, n\text{-}1,n\text{-}2,\ldots,1]$

Best case:

The for loop will run $(n-1)$ time. The while loop will not go inside as $A[i]>key$ will not be satisfied. The algorithm just executes the comparison, so 1 for the line while. All other lines will be executed 1 time within each iteration of the for loop.

Therefore, the total operations for the Best Case can be written as: $(n-1) * (1+1+1+1)$

Worst Case:

Other lines of code inside the for loop have the same number of operations as the previous case, so $(n-1)* (1+1+1)$

Then we look at the "for &while" part together. The while loop will run i times for each $i>0$ as A$[i]>key$ will always be true. Then we have when

$$j=2,\ i=1;$$

$$j=3,\ i=2;$$

$$\ldots.$$

$$j=n,\ i=n-1;$$

Then to combine all the above cases, the total times for the for&while loop is $1+2+\ldots+(n-1) = n(n-1)/2$

There are two lines of code inside the while loop so we can count the total operations as $2*n(n-1)/2$ for them.

Therefore, the total operations for the Worst Case can be written as: $(n-1)* (1+1+1)+ n*(n-1)$

3. UNIX/LINUX Commands

SSH to the school server and practice the following

a) In the Terminal window get a listing of all the files in your home directory. Use

```
ls
ls -a
ls -l
ls -F
```

and observe how the output is different in each case. Get the “man page” for ls and locate the descriptions of the “-a”, “-l”, and “-F” options.

b) Create two subdirectories (of your home directory) called `demo1`, `demo2`.

c) Copy the file “`mdp_phenotype.txt`” (attached under Lab 2 on CourseLink) to demo1 using `scp`.

d) Practice commands `cat, less`, and `more` to check the content of the file.

e) Change your working directory to the `demo2` subdirectory of your home directory. Now using the `cp` command and a pathname involving “..”, copy `mdp_phenotype.txt` from the `demo1` subdirectory to your current directory (`demo2`). Call the new file `new_ mdp_phenotype.txt.`

f) Using either the `diff` or the `cmp` command show that the `new_mdp_phenotype.txt` has the same content `mdp_phenotype.txt` in the `demo1` subdirectory.

g) Under `demo2`, issue the command “`touch newfile.txt`” to create a new, empty file called `newfile.txt`. Use the `ls` or `stat` command with the appropriate option(s) to confirm that that new file’s size is zero.

h) Back to your home directory, using an appropriate combination of the `rm, rmdir` and/or “ `rm -r`” commands, remove those subdirectories and their content. Finally, show that the subdirectories and their content have been deleted.

Solution:

1. Connect to VPN if you are outside of the university network.
2. Connect to the school server using ssh command.
3. Run following commands
   a. `mkdir demo1`
   b. `mkdir demo2`
4. Open new terminal window. Navigate to the folder where mdp_phenotype.txt is located and run scp command
   a. `scp mpd_phenotype.txt [username]@linux.socs.uoguelph.ca:~/demo1/mdp_pheenotype.txt`
5. Go back to the first terminal window and check if the file was successfully transferred (e.g. use ls command)

6. Run `cp ~/demo1/mpd_phenotype.txt ~/demo2/new_mpd_phenotype.txt`
7. Compare files using (there should be no output as files are identical)
    a. `diff ~/demo1/mpd_phenotype.txt ~/demo2/new_mpd_phenotype.txt`
8. Run `touch newfile.txt`
9. Run `ls` (adding -al option allows you to see all files and their properties)
10. Delete files and directories
    a. `rm -r demo1/*`
    b. `rm -r demo2/*`
    c. `rmdir demo1`
    d. `rmdir demo2`

4. Shell Scripts Basics

SSH to the school server and practice the following

a) Using one of the text editors (`nano` or `vim`), create a file named "`login_info`" with the following lines in it

```
#!/bin/sh
echo uptime:
uptime
echo users:
who
```

Create the file in your home directory. Make sure that the very first line is

```
#!/bin/sh
```

Note: Do not start the file with a blank line. Do not start the first line with a space or tab.

b) Determine the functionality of each of the commands in `login_info` by consulting the man page for the command; e.g.

```
man uptime
```

Make sure you understand what each of the commands is supposed to do.

c) Change the permission of the file so that the owner can execute it. Congratulations! You have now created a shell script.

d) Invoke your script by typing the command

```
./login_info
```

e) Delete `login_info`