

Q1

A java code, taking a file into a constructor as well as well as creating a default version

B Learn off some of the Processing functions

C Understand what map and lerp do in processing

Q2

A

understand how Yasc uses **encapsulation, polymorphism, inheritance, interfaces and abstrac**t class and undertand these terms

B Class diagram question

C

for(int I = 0; I < arrayList.size(); i++)

System.out.print(arrayList.get(i))

Or

For(ArrayList type: arrayList)

System.out.print(type)

Q3

a

Files git makes

Gitignore – ignore default files depending on the language selected (eg java ignores all the class files as they are generated from the java files when complied)

License – Help determines who can use and who cannot based upon the license. Also determines what can be done with the software

.git folder – stores all the activities the repo has done, such as all the branches it has work on, the different commit versions stored and all the servers it is connected to. It has a number of subdirectories itself such as logs, info, objects and refs are some of the files and/or subdirectories.

B + C

Git operations

Init – initanse the directory to become a git repo

Clone – clones in an existing repo already on the serve

Diff – compares two versions depending on what you want to do eg if you want compare a file to a index version (staged) to see the difference type git diff if to the latest commit version type git diff HEAD and compare index with commit type git diff –cahched

To compare two commit version type git diff ID1 ID2

To compare two different branches git diff branch1 branch 2

Status

Compare the currenly branch to the same branch on the server to see if there are any change

Add

Adds the files to the staging area

Commit

Commit new files with a id and a log that can tracked from the server

Resets

Resets to a previous commit version

Rm

Removes certain files from the working and/or staging area

Mv

Moves a file from one directory to another

Branch

Works with different branches

Checkout

Moves to another branch, if does not exist it is then made

Merge

Merges one branch into the master or to abort if there is conflicts

MergeToll

Merges certain files together and normally ran with merge

Log

Shows all the different commits

Stash

Saves the file in a diiferent untracked directory

To bring them back do

Apply (keeps the version on the stash)

Pop (deletes the version on the stash)

Tag

Gives each version a tag that can be switched inbetween

Fetch

Fetches all the content of an repo

Pull

pulls all the changes from the server

push

push new changes onto a certain branch on the server

remote

connect the branch to an online branch, depending which you desire

D

Resolve merge conflicts

Type mergeTool –tool=emerge

Follow by n (next confliact)

Type a b c d like a 2 by 2 loop going across then down

Type n until you reach the bottom of all the merges

Type q and click yes or no for “finishing successfully megeing all the files”

Or

Go into the file and delete the undesried changes in between the head and the other end

<<<<

>>>

Q4 study matricx class

1. Operation
2. exceptions
3. levenstein distance for strings (study)