

Homework: 03

16. Technomakers: Anupam Kumar (11940160), Abdur Rahman Khan (11940020), Ruchit Prakash Saxena (11941040)

(4. i)

The git commands are as follows :

```
$ git init
$ touch tree.cpp //code for inputting binary tree
$ echo "< code for input >">>tree.cpp
$ git add .
$ git commit -m "code for taking input"
$ git graph
$ git branch func1
$ git branch func2
$ git branch func3
$ git checkout func1
$ touch preorder.cpp
$ echo "< code for preorder traversal >">>preorder.cpp
$ git add preorder.cpp
$ git graph
$ git checkout func2
$ touch postorder.cpp
$ echo "< code for postorder traversal >">>postorder.cpp
$ git add postorder.cpp
$ git graph
$ git checkout func3
$ touch zigzag.cpp
$ echo "< code for zigzag traversal >">>zigzag.cpp
$ git add zigzag.cpp
$ git graph
$ git checkout master
$ git merge func1 -m"merged func1 to master"
$ git graph
$ git merge func2
$ git graph
$ git merge func3 -m"merged func2 to master"
$ git graph
$ touch input.cpp
$ echo "< code for input/output from user >">>input.cpp
$ git add input.cpp
```

```

$ git graph
$ echo "< updated code for zigzag traversal >">>zigzag.cpp
$ git add zigzag.cpp
$ git graph
$ git checkout -b newFunction
$ touch convesion.cpp
$ echo "< code for converting binarytree to BST >">>convesion.cpp
$ git add convesion.cpp
$ git commit -m "completed code for conversion of BT to BST"
$ git graph

$ git checkout master
$ echo "< updated code for preorder traversal >">>preorder.cpp
$ git add preorder.cpp
$ git commit -m "preorder traversal completely done"
$ echo "< updated code for postorder traversal >">>postorder.cpp
$ git add postorder.cpp
$ git commit -m "postorder traversal completely done"
$ echo "< updated code for zigzag traversal >">>zigzag.cpp
$ git add zigzag.cpp
$ git commit -m "Zigzag traversal completely done"
$ git add input.cpp
$ echo "< updated code for zigzag traversal for reverse zigzag >">>zigzag.cpp
$ git add zigzag.cpp
$ git commit -m "updated it to print reverse zigzag"
$ git add input.cpp
$ git commit -m "added input / output code for user"
$ git checkout newFunction
$ git merge master //merged newFunction with master

```

(4. ii)

The git commands are as follows :

```

$ git init
$ touch tree.cpp //code for inputting binary tree
$ echo "< code for input >">>tree.cpp
$ git add .
$ git commit -m "code for taking input"
$ git graph
$ git branch func1
$ git branch func2
$ git branch func3
$ git checkout func1
$ touch preorder.cpp
$ echo "< code for preorder traversal >">>preorder.cpp
$ git add preorder.cpp
$ git graph
$ git checkout func2

```

```

$ touch postorder.cpp
$ echo "< code for postorder traversal >">>postorder.cpp
$ git add postorder.cpp
$ git graph
$ git checkout func3
$ touch zigzag.cpp
$ echo "< code for zigzag traversal >">>zigzag.cpp
$ git add zigzag.cpp
$ git graph
$ git checkout master
$ git merge func1 -m"merged func1 to master"
$ git graph
$ git merge func2
$ git graph
$ git merge func3 -m"merged func2 to master"
$ git graph
$ touch input.cpp
$ echo "< code for input/output from user >">>input.cpp
$ git add input.cpp
$ git graph
$ echo "< updated code for zigzag traversal >">>zigzag.cpp
$ git add zigzag.cpp
$ git graph
$ git checkout -b NewFunc
$ touch convesion.cpp
$ echo "< code for converting binarytree to BST >">>convesion.cpp
$ git add convesion.cpp
$ git commit -m "completed code for conversion of BT to BST"
$ git graph

```

```

$ git checkout master
$ touch code1.cpp //code for first functionality
$ echo "< code for respective functionality >">>code1.cpp
$ touch code2.cpp //code for second functionality
$ echo "< code for respective functionality >">>code2.cpp
$ touch code3.cpp //code for third functionality
$ echo "< code for respective functionality >">>code3.cpp
$ git checkout NewFunc
$ git add .
$ git commit -m "committed all the changes in newFunc"
$ git checkout master
$ git add code1.cpp
$ git add code2.cpp
$ git rm code3.cpp //remove the third functionality
$ git commit -m "finished code1.cpp and code2.cpp and removed 1 functionality"
$ git rebase NewFunc

```