Q3
commands for gdb
gcc -g error.c // this is the c file provided
gdb a.out // now we can get the gdb prompt and work with it
list // to able to see line number
break 10 // this is the line "ptr=getInfo()"
run // run until breakpoint
print ptr // we can see the pointer is initialized to 0
step // step into the getInfo function
next next // go to the scanf line
// enter the string
print buffer // we can now see the input string in buffer variable
next //go to the printf line
print ptr // we can see the pointer does not change although buffer is changed

reason for the error
getInfo() is returning a local variable that is destroyed by the time it is used by main
getInfo() has to return constant string to fix the error

Q5
Git question
A)
They can do it by creating branch
1.create a remote repository
    git init -bare
2.get the URL
    git config -- get remote.origin.url
3.share it with the other person
4.Now the other person can get the repo by
    git clone <URL>
5.create new branch for p1 and p2
    git branch <name1>
    git branch <name2>
6.p1 and p2 go to their own branch to edit file
    git checkout <name1>
    git checkout <name2>
7.after editing file (for each person)
    git add <filename>
    git commit -m "msg"
    git push <name1>
    git push <name2>
8.after adding them to git
    Go to master and merge the branches

git checkout master

git merge <name1>

git merge <name2>

9.if there is no conflict and everything works out

git branch -d <name1>

git branch -d <name2>

// do not forget to pull master before push else easy to have conflicts

If there is conflict

B)

Have to solve it manually:

Find the conflict and assume use p1's branch

git status // use this to check the status for the conflict

p2 can go to p1's branch : git checkout <name1>

Edit the file

git commit -m "msg"

git push <name1>

Then push branch1 to master and delete files in branch2

git checkout master

git merge <name1>

git branch -d <name1>

git branch -D <name2>

// to avoid conflict pull everytime before making changes

// git pull master before making changes then push to master