**EXPLANATION for R CODE3**

**#In this code, I tried to find the *k*-core values and clustering coefficients of all proteins in each.**

**#protein-protein interaction network. The interaction network is given as a set of undirected edges**

**#that comprise a graph. A *k*-core in a graph is a subgraph in which all the nodes in that subgraph**

**#have at least degree *k*. In other words, each node in a *k*-core has at least *k* immediate neighbors.**

**#Use the O(*n*3) algorithm described in class to find the highest *k*-core a node is a member of**

**# (the value of *k* will be the *k*-core value of that node). I had used my implementation to find**

**#the *k*-core values of the nodes in the following human protein-protein interaction network.**

**R CODE**

code3<- read.table(file.choose()) # this command is for attaching the file.

install.packages("PCIT") # some packages are downloaded that are related with the network analysis.

library(PCIT)

install.packages("network")

library(network)

install.packages("statnet")

update.packages("statnet")

library(statnet)

library(igraph)

code3network<- as.network(code3) #this command convert the data into a network

network.edgecount(code3network) # shows the number of edges that are present

network.size(code3network) #shows the network size

graph<- plot(code3network) #this command will create a visualization of network

adjmatrix<- get.adjacency(graph.edgelist(as.matrix(code3), directed=FALSE)) # the hint was given in the question said using an adjacency matrix/list representation for the PPI network will make things easier.. So, an adjacency matrix was created

kcores<- kcores(graph, mode = "digraph", diag = FALSE, cmode = "freeman", ignore.eval = FALSE) #there exist a code in R programming that calculates kcores from the graph information, that’s why the graph info was found.

#For being sure about the result of the code , the maximum value of k core can be got.

sort(kcores(code3network), decreasing = TRUE) #this command sorts the results from an increasing order.

cc<- clusteringCoefficient(adjmatrix, FUN='localClusteringCoefficient') #this special command is used for finding the clustering coefficient.

sort(cc, decreasing = TRUE)

result<- sort((kcores\*cc),decreasing=TRUE)

🡪Here are the top 10 nodes in the graph with respect to this measure along with its k-core value and clustering coefficient.

Result

PAG1 14 0.41

LCP2 14 0.33

GAB2 14 0.32

GAB1 14 0.29

ZAP 14 0.28

PTPN1114 0.26

EPOR 14 0.26

GHR 14 0.26

CSK 14 0.25

KHDRBS1 14 0.23 #The first column shows the k core value and the second one shows the clustering coefficient.