I know there are more preferred languages, such as Python, C#, Java, JavaScript, Go, Kotlin, C++, Elixir, but it was worth trying.

And at the end, I rewrote everything in Python The same set of instructions continued on the 5th and 6th day, but R should also be just as good as any other language.

The following dataset is, the dataset I have received, your’s will be different. And the code was just a product of 10 minutes writing (hence, not really optimized).

My dataset is presented as:

library(ggplot2)

# read instructions for both wires

d <- c("R1007","D949","R640","D225","R390","D41","R257","D180","L372","U62","L454","U594","L427","U561","R844","D435","L730","U964","L164","U342","R293","D490","L246","U323","L14","D366","L549","U312","L851","U959","L255","U947","L179","U109","R850","D703","L310","U175","L665","U515","R23","D633","L212","U650","R477","U131","L838","D445","R999","D229","L772","U716","L137","U355","R51","D565","L410","D493","L312","U623","L846","D283","R980","U804","L791","U918","L641","U258","R301","U727","L307","U970","L748","U229","L225","U997","L134","D707","L655","D168","L931","D6","R36","D617","L211","D453","L969","U577","R299","D804","R910","D898","R553","U298","L309","D912","R757","U581","R228","U586","L331","D865","R606","D163","R425","U670","R156","U814","L168","D777","R674","D970","L64","U840","L688","U144","L101","U281","L615","D393","R277","U990","L9","U619","L904","D967","L166","U839","L132","U216","R988","U834","R342","U197","L717","U167","L524","U747","L222","U736","L149","D156","L265","U657","L72","D728","L966","U896","R45","D985","R297","U38","R6","D390","L65","D367","R806","U999","L840","D583","R646","U43","L731","D929","L941","D165","R663","U645","L753","U619","R60","D14","L811","D622","L835","U127","L475","D494","R466","U695","R809","U446","R523","D403","R843","U715","L486","D661","L584","U818","L377","D857","L220","U309","R192","U601","R253","D13","L95","U32","L646","D983","R13","U821","L1","U309","L425","U993","L785","U804","L663","U699","L286","U280","R237","U388","L170","D222","L900","U204","R68","D453","R721","U326","L629","D44","R925","D347","R264","D767","L785","U249","R989","D469","L446","D384","L914","U444","R741","U90","R424","U107","R98","U20","R302","U464","L808","D615","R837","U405","L191","D26","R661","D758","L866","D640","L675","U135","R288","D357","R316","D127","R599","U411","R664","D584","L979","D432","R887","D104","R275","D825","L338","D739","R568","D625","L829","D393","L997","D291","L448","D947","L728","U181","L137","D572","L16","U358","R331","D966","R887","D122","L334","D560","R938","D159","R178","D29","L832","D58","R37")

d2 <- c("L993","U121","L882","U500","L740","D222","R574","U947","L541","U949","L219","D492","R108","D621","L875","D715","R274","D858","R510","U668","R677","U327","L284","U537","L371","U810","L360","U333","L926","D144","R162","U750","L741","D360","R792","D256","L44","D893","R969","D996","L905","D524","R538","U141","R70","U347","L383","U74","R893","D560","L39","U447","L205","D783","L244","D40","R374","U507","L946","D934","R962","D138","L584","U562","L624","U69","L77","D137","L441","U671","L849","D283","L742","D459","R105","D265","R312","D734","R47","D369","R676","D429","R160","D814","L881","D830","R395","U598","L413","U817","R855","D377","L338","D413","L294","U321","L714","D217","L15","U341","R342","D480","R660","D11","L192","U518","L654","U13","L984","D866","R877","U801","R413","U66","R269","D750","R294","D143","R929","D786","R606","U816","L562","U938","R484","U32","R136","U30","L393","U209","L838","U451","L387","U413","R518","D9","L847","D605","L8","D805","R348","D174","R865","U962","R926","U401","R445","U720","L843","U785","R287","D656","L489","D465","L192","U68","L738","U962","R384","U288","L517","U396","L955","U556","R707","U329","L589","U604","L583","U457","R545","D504","L521","U711","L232","D329","L110","U167","R311","D234","R284","D984","L778","D295","R603","U349","R942","U81","R972","D505","L301","U422","R840","U689","R225","D780","R379","D200","R57","D781","R166","U245","L865","U790","R654","D127","R125","D363","L989","D976","R993","U702","L461","U165","L747","U656","R617","D115","L783","U187","L462","U838","R854","D516","L978","U846","R203","D46","R833","U393","L322","D17","L160","D278","R919","U611","L59","U709","L472","U871","L377","U111","L612","D177","R712","U628","R858","D54","L612","D303","R205","U430","R494","D306","L474","U848","R816","D104","L967","U886","L866","D366","L120","D735","R694","D335","R399","D198","R132","D787","L749","D612","R525","U163","R660","U316","R482","D412","L376","U170","R891","D202","R408","D333","R842","U965","R955","U440","L26","U747","R447","D8","R319","D188","L532","D39","L863","D599","R307","U253","R22")

Creating two empty dataframes (each for different group) and a helper function to decode the instructions: R1007 -> should be undestood as Right for 1007 steps and in accordance, the coordinates change.

gCoord2 <- function(pos, prevX, prevY){

direction <- substring(pos,1,1)

val <- as.integer(substring(pos,2,nchar(pos)))

#a <- c(0,0)

if (direction == "R") {

a <- c(prevX,prevY+val)

return(a)}

if (direction == "L") {

a <- c(prevX,prevY-val)

return(a)}

if (direction == "U") {

a <- c(prevX+val,prevY)

return(a)}

if (direction == "D") {

a <- c(prevX-val, prevY)

return(a)}

}

And not to iterate through both datasets and binding them into single one:

df <- data.frame(x = c(0), y = c(0), group = 1)

df2 <- data.frame(x = c(0), y = c(0), group = 2)

for (i in 1:length(d)){

ii <- d[i]

print(ii)

#get last value from df

X <- tail(df$x,1)

Y <- tail(df$y,1)

x1<- gCoord2(d[i],X,Y)[1]

y1<- gCoord2(d[i],X,Y)[2]

df <-rbind(df, c(x=x1, y=y1, group=1))

}

for (i in 1:length(d2)){

ii <- d2[i]

print(ii)

#get last value from df

X <- tail(df2$x,1)

Y <- tail(df2$y,1)

x1<- gCoord2(d2[i],X,Y)[1]

y1<- gCoord2(d2[i],X,Y)[2]

df2 <-rbind(df2, c(x=x1, y=y1, group=2))

}

df3 <- rbind(df,df2)

Finally, touch of ggplot2:

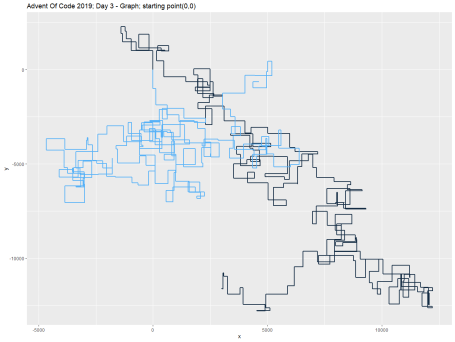
ggplot(df3, aes(x = x, y = y, group = group, colour=group)) +

geom\_path(size = 0.75, show.legend = FALSE) +

# theme\_void() +

ggtitle('Advent Of Code 2019; Day 3 - Graph; starting point(0,0)')

And the graph with both wires:



Happy R-coding!