IS 471: Spring 2018

Homework 4

Please write your code in R and submit your script and results for each of the following.

1. Store the following binary number as a character string s: “110011”. (10 points)

library(seqinr)  
library(compositions)  
s = "110011"

1. Convert this into a character vector v. (10 points)

v = unlist(strsplit(s, "")) # "1" "1" "0" "0" "1" "1"

1. Expand the character vector v to 8 bits as follows: swap v[3] and v[4] . (10 points)

swap(v[3], v[4]) #   
v = unlist(strsplit(s, "")) # "1" "1" "0" "0" "1" "1"

1. Replicate v[3] and v[4] . The result (call it ev) should be a character vector of size 8. (10 points)

ev = vector(mode="character", length = 8)  
ev = append(v[1:6], v[3:4], after = 2) # # "1" "1" "0" "0" "0" "0" "1" "1"

1. Write a function called expand() that takes a binary vector of size 6 as input and returns a binary vector of size 8 as output. (20 points)

expand = function(binaryVec){  
   
 v = unlist(strsplit(binaryVec, ""))  
 swap(v[3], v[4])  
 ev = vector(mode="character", length = 8)  
 ev = append(v[1:6], v[3:4], after = 2)  
 return(ev)  
}  
expand("123456") # "1" "2" "4" "3" "4" "3" "5" "6"

## [1] "1" "2" "4" "3" "4" "3" "5" "6"

1. Create two character vectors S11 and S12 such that S11 contains the binary representations of the numbers (5,2,1,6,3,4,7,0), and S12 contains those of the numbers (1,4,6,2,0,7,5,3). (10 points)

numbers = c(5,2,1,6,3,4,7,0)  
s11=binary(numbers) # [1] "101" "010" "001" "110" "011" "100" "111" "000"  
  
numbers2 = c(1,4,6,2,0,7,5,3)  
s12=binary(numbers2) # [1] "001" "100" "110" "010" "000" "111" "101" "011"

1. Let b=“1101”. Extract the last three characters. Convert the last three characters into decimal and store it in a variable called tempVal. Write an “if-then-else” statement such that if b[1] == 0 retrieve the value of S11[tempval]. If b[1]==1 retrieve the value of S12[tempval] and store it in a variable tempRet. (10 points)

b=unlist(strsplit("1101", ""))  
tempVal=b  
tempVal=c(tempVal[2:4])  
tempVal=paste(tempVal[1:3])  
tempVal=toString(c(tempVal[1], tempVal[2], tempVal[3]))  
tempVal=gsub(pattern = ", ", replacement = "", x = tempVal, fixed = TRUE)  
tempVal=strtoi(tempVal, base=2)  
  
if (b[1]==0) {  
 tempRet= s11[tempVal+1]  
} else {  
 tempRet= s12[tempVal+1]  
}  
#111

1. Write a function called S11\_function that takes a block of 4-character binary (such as “1110”) and, using the two s-boxes from question 6, returns a 3-character binary number using the logic that if the first character is “0” then use S11, otherwise use S12. (20 points)

s11\_function = function(binary\_val){  
 b=unlist(strsplit(binary\_val, ""))  
 tempVal=b  
 tempVal=c(tempVal[2:4])  
 tempVal=paste(tempVal[1:3])  
 tempVal=toString(c(tempVal[1], tempVal[2], tempVal[3]))  
 tempVal=gsub(pattern = ", ", replacement = "", x = tempVal, fixed = TRUE)  
 tempVal=strtoi(tempVal, base=2)  
   
 if (b[1]==0) {  
 tempRet= s11[tempVal+1]  
 } else {  
 tempRet= s12[tempVal+1]  
 }  
   
 return(tempRet)  
}  
s11\_function("1000") #001

## [1] "001"