```julia
## IO.jl

# Read an instance of the Clever Traveling Salesperson Problem
# Input: filename = path + filename of the instance
function read_instance(filename)
    # Opens the file
    f = open(filename)
    # Reads the name of the instance
    name = split(readline(f))[2]
    # reads the upper bound value
    upper_bound = parse(Int64,split(readline(f))[2])
    readline(f)#type
    readline(f)#comment
    # reads the dimentions of the problem
    dimention = parse(Int64,split(readline(f))[2]);
    readline(f)#Edge1
    readline(f)#Edge2
    readline(f)#Edge3
    readline(f)#Dimention 2

    # Initialises the cost matrix
    cost = zeros(Int64,dimention,dimention)
    # Reads the cost matrix
    for i in 1:dimention
        data = parse.(Int64,split(readline(f)))
        cost[i,:]=data
    end

    # Closes the file
    close(f)

    # Returns the input data
    return name, upper_bound, dimention, cost
end

function writeSolution(solution, solutionLocation)
    wDir = string(pwd())

    dir, file = splitdir(solutionLocation)
    if (!isdir(dir))
        mkpath(string("./", dir, "/"))
    end

    open(string(wDir, "/", solutionLocation), "w") do f
        for i in eachindex(solution)
            write(f, string(solution[i]-1, " "))
        end
    end
end
```