Pseudocode for Project 1 – Isolation Self-Assessment Tool

Your Name (*first last*): William Cooper

Function main(Arguments argc, argv)

String a = "yes"

String b = "no"

String user\_input = ""

Integer dayPositive, monthPositive, yearPositive

Integer dayExposed, monthExposed, yearExposed

Integer daySecondDose, monthSecondDose, yearSecondDose

Output "Have you tested positive (yes/no)? "

Input user\_input

If user\_input equals a Then

Output "What date did you test positive? "

Input monthPositive, dayPositive, yearPositive

Date exposed = Create Date(dayPositive, monthPositive, yearPositive)

Date exposed2 = Create Date(dayPositive + 5, monthPositive, yearPositive)

Output "You must quarantine from " + exposed.showDate() + " to " + exposed2.showDate()

Else If user\_input equals b Then

Output "Were you exposed (yes/no)? "

Input user\_input

If user\_input equals a Then

Output "Have you received both vaccines (yes/no)? "

Input user\_input

If user\_input equals a Then

Output "When did you receive your second vaccine? "

Input monthSecondDose, daySecondDose, yearSecondDose

Output "When were you exposed? "

Input monthExposed, dayExposed, yearExposed

Date vaccine1 = Create Date(daySecondDose, monthSecondDose, yearSecondDose)

Date exposed1 = Create Date(dayExposed, monthExposed, yearExposed)

If calcDays(vaccine1, exposed1) < 14 Then

Output "You have only been vaccinated for " + calcDays(vaccine1, exposed1) + " days."

Output "You must quarantine for 10 days."

Else

Output "You have been vaccinated for " + calcDays(vaccine1, exposed1) + " days."

Output "You must quarantine for 5 days."

End If

End If

If user\_input equals b Then

Output "You must quarantine for 10 days."

End If

End If

If user\_input equals b Then

Output "You do not need to quarantine."

End If

End If

Output (a equals b)

Return 0

End Function  
  
  
  
  
Class Date

Integer day, month, year

Function Date()

day = 1

month = 1

year = 2021

End Function

Function Date(Integer d, Integer m, Integer y)

If y >= 2021 And y <= 2022 Then

year = y

Else

year = 2021

End If

If m >= 1 And m <= 12 Then

month = m

Else

month = 1

End If

day = d

End Function

Function setDay(Integer d) -> Boolean

If d >= 0 Then

day = d

Return True

Else

day = 1

Return False

End If

End Function

Function setMonth(Integer m) -> Boolean

If m >= 0 Then

month = m

Return True

Else

month = 1

Return False

End If

End Function

Function setYear(Integer y) -> Boolean

If y >= 0 Then

year = y

Return True

Else

year = 1

Return False

End If

End Function

Function getDay() -> Integer

Return day

End Function

Function getMonth() -> Integer

Return month

End Function

Function getYear() -> Integer

Return year

End Function

Function showDate() -> String

String dayStr, monthStr, yearStr

If day < 10 Then

dayStr = "0" + ToString(day)

Else

dayStr = ToString(day)

End If

If month < 10 Then

monthStr = "0" + ToString(month)

Else

monthStr = ToString(month)

End If

yearStr = ToString(year)

Return monthStr + "/" + dayStr + "/" + yearStr

End Function

End Class  
  
  
  
  
  
 Function calcDays(a: Date, b: Date) -> Integer

Constants:

daysInYear = 365

baseYear = 2021

daysPerMonth[] = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]

Variables:

totalDaysA = 0

totalDaysB = 0

For year in range(baseYear to a.getYear() exclusive)

totalDaysA = totalDaysA + daysInYear

End For

For year in range(baseYear to b.getYear() exclusive)

totalDaysB = totalDaysB + daysInYear

End For

For month in range(1 to a.getMonth() exclusive)

totalDaysA = totalDaysA + daysPerMonth[month - 1]

End For

For month in range(1 to b.getMonth() exclusive)

totalDaysB = totalDaysB + daysPerMonth[month - 1]

End For

totalDaysA = totalDaysA + a.getDay()

totalDaysB = totalDaysB + b.getDay()

daysDifference = totalDaysB - totalDaysA

Return daysDifference

End Function