Warsaw University of Technology

Programming in Logic and Symbolic Programming

Maleden Expert System

Authors:
Abba Umar

January 15, 2023

Contents

1 Expert System

1.1 Project description

Maladen(Malaria and Dengue) is an expert system that consider facts and symptoms of mosquito virus to provide diagnosis to patient regarding the specific disease they have. This implies that maladen expert system uses knowledge about mosquito virus and fact about patients to suggest whether the person have malaria or dengue.

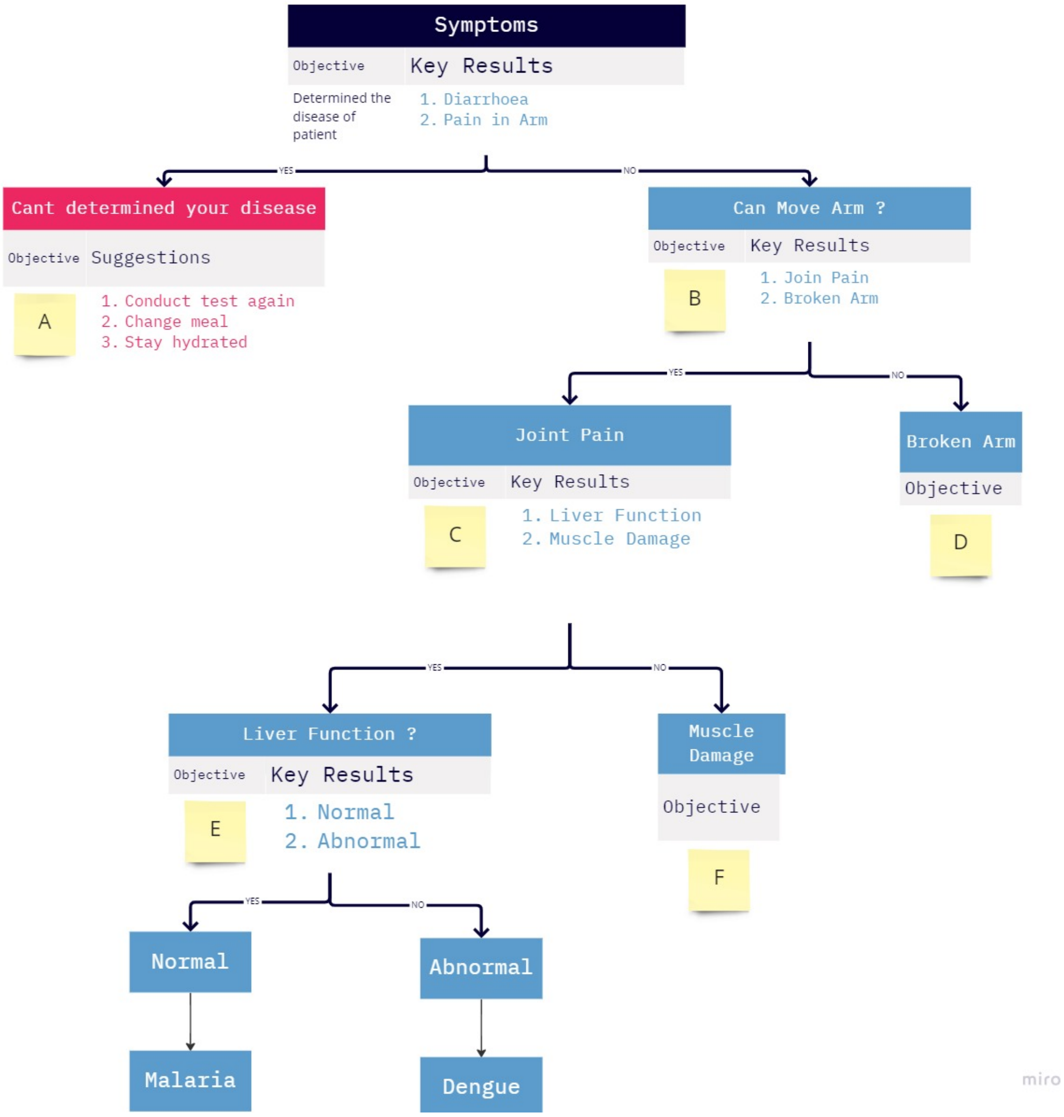

Figure 1: Maladen Decision Tree

1.2 Code

```
;make the first letter of the result to capital letter
(setq *print-case* :capitalize)
; set the diagnosis to nill initially
(defvar *diagnosis* nil)
;macro to print question to ask patient
(defmacro basic-print(name)
(format t ”Enter your  a:” name))
;function to get patient name
(defun get-name()
(basic-print ”name”)
(defvar *name* (read)))
(get-name)
;function to get patient age
(defun get-age()
(basic-print ”age”)
(defvar *age* (read-line)))
(get-age)
;function to get patient gender
(defun get-gender()
(basic-print ”gender”)
(defvar *gender* (read-line)))
(get-gender)
;function to get patient complaint
(defun complaint()
(princ ”Enter complaints of patient:”)
(defvar *pain* (read)))
```

```
(complaint)
;function to get where patient pain is present
(defun presence()
(princ "When is the pain present?:")
(defvar *present* (read)))
(presence)
;function to get patient liver function status
(defun liver-function()
(princ "How does your liver function?:")
(defvar *liver* (read)))
(liver-function)
(terpri)
(format t "Patient Personal Information
(terpri)
(format t "Dear  a,  a years old and gender is  a
(terpri)
;macro to tell diagnosis result of patient
(defmacro diagnosis(disease)
(format t "You are suffering from  a" disease))
;macro to compare and get condition about what patient enter
(defmacro compare (condition rest body)
`(if ,condition (progn ,@body) (if ,condition (progn ,@body) )))
;function to compare patient entries
(defun main()
(cond ( (and (equal *pain* 'arm-pain) (equal *present* 'moving)) ; If T do this
(compare (equal *liver* 'normal)
(setf *diagnosis* 'malaria)
```

```
(print ”You are suffering from Malaria”)
(terpri)))
( (and (equal *pain* ’arm-pain) (equal *present* ’resting)) ; Else If T do this (compare (equal *liver* ’abnormal)
(setf *diagnosis* ’dengue)
(print ”You are suffering from Dengue”)
(terpri)))
(t (format t ”Don’t Know
(main)
;macro to suggest to patient mitigations
(defmacro suggester (rest clauses)
(if (consp clauses)
(destructuring-bind ((pred . forms) . rest-clauses) clauses
‘(if ,pred (progn ,@forms)
,(if (and (consp rest-clauses) (eq (caar rest-clauses) ’t))
‘(progn ,@(cdar rest-clauses))
‘(suggester ,@rest-clauses)))) nil))
(suggester ((equal *diagnosis* ’dengue) (write ”You should avoid mosquito bites to reduce risk of further transmission”))
((equal *diagnosis* ’malaria) (write ”Please do not sleep in open air and cover your full skin Because”))
(t (write ”I cant suggest anything”)))
```