

Proposta A: Diagnóstico e tratamento de doenças em peixes

```
fish.disease <- function(){ #criando uma função de nome fish.disease
#####
#####
#
#
# A seguinte função gera o diagnóstico e tratamento para doenças em
peixes a partir #
# dos principais sintomas apresentados pelo animal. Os sintomas são
fornecidos pelo #
# usuário por meio de respostas as perguntas geradas pela função.A função
foi #
# elaborada com base no fluxograma disponibilizado pela empresa Alcon no
site #
# http://alconpet.com.br/home, levando em consideração 5 medicamentos:
Alcali, Acid, #
# Aqualife, e Bacter.
#
#
#
#
#
#
#
#
#
#
#####
#####
##### coletando dados da altura do aquário
#####
h <- as.numeric(readline("\tQual a altura do aquário em que o(s) peixe(s)
está(ão), em cm? \n\tDigite apenas números maiores do que zero (ex.:35):
\n"))#coletando o dado de altura do aquário através de interação com o
usuário e transformando em um vetor da classe numérica
#enquanto o valor de altura (h) for menor do que zero, a função deve
manter o loop que testa se h é um número positivo e pedir que outro valor de
h seja dado
while (h < 0){
cat("\tA altura do aquário precisa ser um número positivo, em cm.\n\tPor
favor, não introduza caracteres diferentes de números.\n\tApós digitar o
número, aperte enter.\n") #Alertando o usuário para a digitação de número
inválido para altura do aquário)
h <-as.numeric(readline("\tQual a altura do aquário em que o(s) peixe(s)
está(ão), em cm? \n\tDigite apenas números maiores do que zero (ex.:35):
\n"))#coletando dado de altura do aquário através de interação com o usuário
}
##### coletando dados da largura do aquário
#####
L <- as.numeric(readline("\tQual a largura do aquário em que o(s) peixe(s)
está(ão), em cm? \n\tDigite apenas números maiores do que zero (ex.:30):
```

```
\n"))#coletando o dado de largura do aquário através de interação com o
usuário e transformando em um vetor da classe numérica
#enquanto o valor de largura (L) for menor do que zero, a função deve
manter o loop que testa se L é um número positivo e pedir que outro valor de
L seja dado
while (L < 0){
  cat("\tA largura do aquário precisa ser um número positivo, em
cm.\n\tPor favor, não introduza caracteres diferentes de números.\n\tApós
digitar o número, aperte enter.\n") #Alertando o usuário para a digitação de
número inválido para largura do aquário
  L <- as.numeric(readline("\tQual a largura do aquário em que o(s)
peixe(s) está(ão), em cm? \n\tDigite apenas números maiores do que zero
(ex.:30): \n"))#coletando dado de largura do aquário através de interação
com o usuário e transformando em um vetor da classe numérica
}
##### coletando dados de comprimento do aquário
#####
comp <- as.numeric(readline("\tQual o comprimento do aquário em que o(s)
peixe(s) está(ão), em cm? \n\tDigite apenas números maiores do que zero
(ex.:50): \n"))#coletando o dado de comprimento do aquário através de
interação com o usuário e transformando em um vetor da classe numérica
#enquanto o valor de comprimento do aquário (comp) for menor do que zero,
a função deve manter o loop que testa se comp é um número positivo e pedir
que outro valor de comp seja dado
while (comp < 0){
  cat("\tA largura do aquário precisa ser um número positivo, em
cm.\n\tPor favor, não introduza caracteres diferentes de números.\n\tApós
digitar o número, aperte enter.\n") #Alertando o usuário para a digitação de
número inválido para largura do aquário
  comp <- as.numeric(readline("\tQual o comprimento do aquário em que o(s)
peixe(s) está(ão), em cm? \n\tDigite apenas números maiores do que zero
(ex.:50): \n"))#coletando dado de comprimento do aquário através de
interação com o usuário e transformando em um vetor da classe numérica
}
##### Cálculo do volume do aquário em litros
#####
vol <- (h*L*comp)/1000.0 #cálculo do volume do aquário em litros,
considerando que este seja retangular ou quadrado e que as medidas
fornecidas estejam em cm
##### Transposição do fluxograma em código
#####
respl <- readline("\t0 peixe apresenta manchas ou estruturas anormais na
superfície do corpo? \n\tDigite s para sim e n para não: \n")#coletando
dado sobre sintoma do peixe através de interação com o usuário
#enquanto o dado digitado pelo usuário for diferente de s ou n, a função
deve manter o loop que testa se respl é diferente de s ou n e pedir que
outro caracter seja fornecido
while (respl!="s" & respl!= "n"){
  cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para não.
\n\tApós digitar, aperte enter.\n") #Alertando o usuário para a digitação de
```

respostas válidas para o sintoma.

```
resp1 <- readline("\t0 peixe apresenta manchas ou estruturas anormais na
superfície do corpo. \n\tDigite s para sim e n para não: \n")#coletando
dado sobre sintoma do peixe através de interação com o usuário
```

```
}
```

```
#Se a resposta armazenada em resp(n) for sim (s) a função "entra" nesse
if,se não for entra no else
```

```
if(resp1=="s"){
```

```
    resp2 <- readline("\tEstas estruturas são pontos visíveis na pele?
\n\tDigite s para sim e n para não: \n")#coletando dado sobre sintoma do
peixe através de interação com o usuário
```

```
#enquanto o dado digitado pelo usuário for diferente de s ou n, a função
deve manter o loop que testa se resp2 é diferente de s ou n e pedir que
outro caracter seja fornecido
```

```
while (resp2!="s" & resp2!= "n"){
```

```
    cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para
não.\n\t Após digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.
```

```
    resp2 <- readline("\tEstas estruturas são pontos visíveis na pele?
\n\tDigite s para sim e n para não: \n")#coletando dado sobre sintoma do
peixe através de interação com o usuário
```

```
}
```

```
#Se a resposta armazenada em resp(n) for sim (s) a função "entra" nesse
if,se não for entra no else
```

```
if(resp2=="s"){
```

```
    resp3 <- readline("\tSão pequenos pontos brancos espalhados por todo o
corpo e nadadeiras? \n\tDigite s para sim e n para não: \n")#coletando dado
sobre sintoma do peixe através de interação com o usuário
```

```
#enquanto o dado digitado pelo usuário for diferente de s ou n, a
função deve manter o loop que testa se resp3 é diferente de s ou n e pedir
que outro caracter seja fornecido
```

```
while (resp3!="s" & resp3!= "n"){
```

```
    cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para
não.\n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.
```

```
    resp3 <- readline("\tSão pequenos pontos brancos espalhados por todo
o corpo e nadadeiras? \n\tDigite s para sim e n para não: \n")#coletando
dado sobre sintoma do peixe através de interação com o usuário
```

```
}
```

```
#Se a resposta armazenada em resp(n) for sim (s) a função "entra"
nesse if ,se não for entra no else
```

```
if(resp3=="s"){
```

```
    #####doença dos pontos brancos, Ictio
```

```
    print("Doença dos Pontos Brancos")
```

```
    #####Cálculo da dosagem do tratamento com Ictio ou Aqualife: 1 gota
para 2 litros.
```

```
    gotas_Ic_Aq <- vol/2.0
```

```
    #####resposta da função a esse conjunto de sintomas:
```

```
    return(cat("\tSeu(s) peixe(s) apresenta(m) Doença dos pontos Brancos
ou Ictio. \n\tEla é causada pelo protozoário Ichthyophthyrus multifiliis.
\n\tOutros sintomas: Nadadeiras fechadas, peixe se esfregando nas pedras
```

```
\n\te no fundo do aquário, dificuldade para respirar. \n\tTratamento:
Adicione ",gotas_Ic_Aq, " gota(s) de Ictio na água do aquário.\n") )
    }else {
        resp4 <- readline("\tOs pontos são brilhantes, numerosos, similares
a gotinhas douradas aveludadas? \n\tDigite s para sim e n para não:
\n")#coletando dado sobre sintoma do peixe através de interação com o
usuário
        #enquanto o dado digitado pelo usuário for diferente de s ou n, a
função deve manter o loop que testa se resp4 é diferente de s ou n e pedir
que outro caracter seja fornecido
        while (resp4!="s" & resp4!= "n"){
            cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para
não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.
            resp4 <- readline("\tOs pontos são brilhantes, numerosos,
similares a gotinhas douradas aveludadas? \n\tDigite s para sim e n para
não: \n")#coletando dado sobre sintoma do peixe através de interação com o
usuário
        }
        #Se a resposta armazenada em resp(n) for sim (s) a função "entra"
nesse if,se não for entra no else
        if(resp4=="s"){
            ###doença do veludo ou Oodiniose, Ictio+Aqualife
            print("Doença do Veludo\n")
            ###Cálculo da dosagem do tratamento com Ictio e Aqualife: 1 gota
de aqualife para para 4 litros e 1 gota de Ictio para 2 litros
            #####(como há interação medicamentosa, a dosagem de cada
medicamento varia quando usada em conjunto)
            gotas_Aq <- vol/4.0
            gotas_Ic <- vol/2.0
            #####resposta da função a esse conjunto de sintomas:
            return(cat("\tSeu(s) peixe(s) apresenta(m) Doença do Veludo ou
Oodiniose. \n\tEla é causada pelo protozoário Oodinium pilularis.
\n\tOutros sintomas: Falta de apetite, respiração ofegante, perda do
equilíbrio \n\te peixes esbranquiçados. \n\tTratamento: Adicione ",
gotas_Ic, "gota(s) de Ictio para ", gotas_Aq," gota(s) de Aqualife na água
do aquário." ))
        }else {
            ### fim da linha.
            ### Há algumas combinações de sintomas que não levam a nenhum
diagnóstico. Nesse casos, a solução que encontrei foi direcionar o usuário a
refazer
            ### o teste de modo a repensar se os sintomas que ele apontou
anteriormente estavam certos:
            respF <- readline("\tNão foi possível identificar a doença que
acomete seu peixe por meio dos sintomas informados.\n\tVamos refazer o
teste? \n\tDigite s para sim e n para não: \n")#Verificando se o usuário
deseja refazer o teste através de interação com o usuário
            #enquanto o dado digitado pelo usuário for diferente de s ou n, a
função deve manter o loop que testa se respF é diferente de s ou n e pedir
```

```
que outro caracter seja fornecido
    while (respF!="s" & respF!= "n"){
        cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para
não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o respF.
        respF <- readline("\tNão foi possível identificar a doença que
acomete seu peixe por meio dos sintomas informados.\n\tVamos refazer o
teste? \n\tDigite s para sim e n para não: \n")#Verificando se o usuário
deseja refazer o teste através de interação com o usuário
    }
    #Se a resposta armazenada em respF for sim (s) a função "entra"
nesse if,se não for entra no else
    if(respF=="s"){
        cat("\tOk, vamos recomeçar o teste.\n")
        fish.disease()#chamando a função dentro da função para recomeçar
o teste
    }else {
        ###Fim da linha
        return(cat("\tOk. Que pena!\n\tReavalie os sintomas apresentados
pelo seu animal \n\tnovamente e, se necessário, refaça o teste. \n"))
    }
}
}
}else {
    resp3 <- readline("\tHá uma crosta aparentando limo ou manchas como
mofo? \n\tDigite s para sim e n para não: \n")#coletando dado sobre sintoma
do peixe através de interação com o usuário
    #enquanto o dado digitado pelo usuário for diferente de s ou n, a
função deve manter o loop que testa se resp3 é diferente de s ou n e pedir
que outro caracter seja fornecido
    while (resp3!="s" & resp3!= "n"){
        cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para não.
\n\tApós digitar, aperte enter.\n") #Alertando o usuário para a digitação de
respostas válidas para o sintoma.
        resp3 <- readline("\tHá uma crosta aparentando limo ou manchas como
mofo? \n\tDigite s para sim e n para não: \n")#coletando dado sobre sintoma
do peixe através de interação com o usuário
    }
    #Se a resposta armazenada em resp(n) for sim (s) a função "entra"
nesse if ,se não for entra no else
    if(resp3=="s"){
        resp4 <- readline("\tHá um limo branco acizentado, cobrindo as
brânquias e grande extensão do corpo? \n\tDigite s para sim e n para não:
\n")#coletando dado sobre sintoma do peixe através de interação com o
usuário
        #enquanto o dado digitado pelo usuário for diferente de s ou n, a
função deve manter o loop que testa se resp4 é diferente de s ou n e pedir
que outro caracter seja fornecido
        while (resp4!="s" & resp4!= "n"){
            cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para
não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
```

digitação de respostas válidas para o sintoma.

```
resp4 <- readline("\tHá um limo branco acizentado, cobrindo as
brânquias e grande extensão do corpo? \n\tDigite s para sim e n para não:
\n")#coletando dado sobre sintoma do peixe através de interação com o
usuário
}
#Se a resposta armazenada em resp(n) for sim (s) a função "entra"
nesse if,se não for entra no else
if(resp4=="s"){
  ###Costiose, Ictio+Aqualife
  print("Costiose")
  ###Cálculo da dosagem do tratamento com Ictio e Aqualife: 1 gota
de aqualife para para 4 litros e 1 gota de Ictio para 2 litros
  #####(como há interação medicamentosa, a dosagem de cada
medicamento varia quando usada em conjunto)
  gotas_Aq <- vol/4.0
  gotas_Ic <- vol/2.0
  ####resposta da função a esse conjunto de sintomas:
  return(cat("\tSeu(s) peixe(s) apresenta(m) Costiose. \n\tEla
causada pelos protozoários Ichtyobodo sp. (Costia sp.) e Chilodonella sp..
\n\tOutros sintomas: Falta de apetite, ramificações vermelhas nas
nadadeiras, \n\tpeixe se esfregando no fundo do aquário. \n\tTratamento:
Adicione ", gotas_Ic," gota(s) de Ictio para ",gotas_Aq," gota(s) de
Aqualife na água do aquário.\n"))
}else {
  resp5 <- readline("\tO limo é nebuloso e cobre apenas os olhos?
\n\tDigite s para sim e n para não: \n")#coletando dado sobre sintoma do
peixe através de interação com o usuário
  #enquanto o dado digitado pelo usuário for diferente de s ou n, a
função deve manter o loop que testa se resp5 é diferente de s ou n e pedir
que outro caracter seja fornecido
  while (resp5!="s" & resp5!="n"){
    cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para
não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.
    resp5 <- readline("\tO limo é nebuloso e cobre apenas os olhos?
\n\tDigite s para sim e n para não: \n")#coletando dado sobre sintoma do
peixe através de interação com o usuário
  }
  #Se a resposta armazenada em resp(n) for sim (s) a função "entra"
nesse if,se não for entra no else
  if(resp5=="s"){
    resp6 <- readline("\tOs olhos se mostram saltados? \n\tDigite s
para sim e n para não: \n")#coletando dado sobre sintoma do peixe através
de interação com o usuário
    #enquanto o dado digitado pelo usuário for diferente de s ou n,
a função deve manter o loop que testa se resp6 é diferente de s ou n e
pedir que outro caracter seja fornecido
    while (resp6!="s" & resp6!="n"){
      cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para
```

```

não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.
    resp6 <- readline("\tOs olhos se mostram saltados? \n\tDigite
s para sim e n para não: \n")#coletando dado sobre sintoma do peixe através
de interação com o usuário
    }
    #Se a resposta armazenada em resp(n) for sim (s) a função
"entra" nesse if ,se não for entra no else
    if(resp6=="s"){
        ###olhos inchados, Ictio+Bacter
        print("Olhos Inchados")
        #Cálculo da dosagem do tratamento com Ictio ou Aqualife: 1
gota para 2 litros.
        gotas_Ic_Aq <- vol/2.0
        #Cálculo da dosagem do tratamento com Bacter: 1 cápsula para
cada 12,5 litros.
        capsulas <- vol/12.5
        ####resposta da função a esse conjunto de sintomas:
        return(cat("\tSeu(s) peixe(s) apresenta(m) Olhos Inchados
(Pop-eye). \n\tÉ causada por bactérias, fungos ou vermes. \n\tOutros
sintomas: Barriga inchada, nadadeiras roídas. \n\tTratamento: Adicione
",gotas_Ic_Aq," gota(s) de Ictio para ",capsulas," cápsulas(s) de Bacter na
água do aquário.\n"))
    }else {
        ####olhos embaçados, Ictio+Aqualife
        print("Olhos Embaçados")
        ###Cálculo da dosagem do tratamento com Ictio e Aqualife: 1
gota de aqualife para para 4 litros e 1 gota de Ictio para 2 litros
        #####(como há interação medicamentosa, a dosagem de cada
medicamento varia quando usada em conjunto)
        gotas_Aq <- vol/4.0
        gotas_Ic <- vol/2.0
        ####resposta da função a esse conjunto de sintomas:
        return(cat("\tSeu(s) peixe(s) apresenta(m) Olhos Embaçados.
\n\tEssa doença é causada pelos trematodos monogenéticos Dactylogyrus sp. ou
Gyrodactylus sp.. \n\tOutros sintomas: Falta de apetite, respiração ofegante
\n\te inchaço das brânquias. \n\tTratamento: Adicione ",gotas_Ic," gota(s)
de Ictio para ",gotas_Aq," gota(s) de Aqualife na água do aquário.\n"))
    }
    }else {
        resp6 <- readline("\tO peixe apresenta muco excessivo ao redor
da boca chegando a formar tufos semelhantes a algodão? \n\tDigite s para sim
e n para não: \n")#coletando dado sobre sintoma do peixe através de
interação com o usuário
        #enquanto o dado digitado pelo usuário for diferente de s ou n,
a função deve manter o loop que testa se resp6 é diferente de s ou n e
pedir que outro caracter seja fornecido
        while (resp6!="s" & resp6!="n"){
            cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para
não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.

```

```
resp6<- readline("\t0 peixe apresenta muco excessivo ao redor
da boca chegando a formar tufos semelhantes a algodão? \n\tDigite s para sim
e n para não: \n")#coletando dado sobre sintoma do peixe através de
interação com o usuário
}
#Se a resposta armazenada em resp(n) for sim (s) a função
"entra" nesse if ,se não for entra no else
if(resp6=="s"){
  ###fungo na boca, Bacter+aqualife
  print("Fungo na Boca")
  #Cálculo da dosagem do tratamento com Ictio ou Aqualife: 1
gota para 2 litros.
  gotas_Ic_Aq <- vol/2.0
  #Cálculo da dosagem do tratamento com Bacter: 1 cápsula para
cada 12,5 litros.
  capsulas <- vol/12.5
  ####resposta da função a esse conjunto de sintomas:
  return(cat("\tSeu(s) peixe(s) apresenta(m) Fungo na Boca
(Cotton mouth). \n\tApesar do nome, normalmente é causada pela ação de
bactérias Flexibacter columnaris. \n\tÉ comum a ocorrência simultânea de
fungos. \n\tTratamento: Adicione ",gotas_Ic_Aq," gota(s) de Aqualife para
",capsulas," cápsulas(s) de Bacter na água do aquário.\n"))
}else {
  resp7 <- readline("\tExistem manchas brancas ou tufos
semelhantes a algodão no corpo? \n\tDigite s para sim e n para não:
\n")#coletando dado sobre sintoma do peixe através de interação com o
usuário

  #enquanto o dado digitado pelo usuário for diferente de s ou
n, a função deve manter o loop que testa se resp7 é diferente de s ou n e
pedir que outro caracter seja fornecido
  while (resp7!="s" & resp7!="n"){
    cat("\tDigite apenas s minúsculo para sim, ou n minúsculo
para não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.
    resp7 <- readline("\tExistem manchas brancas ou tufos
semelhantes a algodão no corpo? \n\tDigite s para sim e n para não:
\n")#coletando dado sobre sintoma do peixe através de interação com o
usuário
  }
  #Se a resposta armazenada em resp(n) for sim (s) a função
"entra" nesse if ,se não for entra no else
  if(resp7=="s"){
    ###Doença do algodão, Aqualife
    print("Doença do Algodão")
    #Cálculo da dosagem do tratamento com Ictio ou Aqualife: 1
gota para 2 litros.
    gotas_Ic_Aq <- vol/2.0
    ####resposta da função a esse conjunto de sintomas:
    return(cat("\tSeu(s) peixe(s) apresenta(m) Doença do
Algodão. \n\tÉ causada pelos fungos Saprolegnia sp., Achlya sp. ou
```

```
Ichthyosporidium sp. \n\tOutros sintomas: Perda de escamas. \n\tTratamento:  
Adicione ",gotas_Ic_Aq," gota(s) de Aqualife na água do aquário.\n"))  
    }else {  
        ### fim da linha.  
        ### Há algumas combinações de sintomas que não levam a  
nenhum diagnóstico. Nesse casos, a solução que encontrei foi direcionar o  
usuário a refazer  
        ### o teste de modo a repensar se os sintomas que ele  
apontou anteriormente estavam certos:  
        respF <- readline("\tNão foi possível identificar a doença  
que acomete seu peixe por meio dos sintomas informados.\n\tVamos refazer o  
teste? \n\tDigite s para sim e n para não: \n")#Verificando se o usuário  
deseja refazer o teste através de interação com o usuário  
        #enquanto o dado digitado pelo usuário for diferente de s ou  
n, a função deve manter o loop que testa se respF é diferente de s ou n e  
pedir que outro caracter seja fornecido  
        while (respF!="s" & respF!= "n"){  
            cat("\tDigite apenas s minúsculo para sim, ou n minúsculo  
para não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a  
digitação de respostas válidas para o respF.  
            respF <- readline("\tNão foi possível identificar a doença  
que acomete seu peixe por meio dos sintomas informados.\n\tVamos refazer o  
teste? \n\tDigite s para sim e n para não: \n")#Verificando se o usuário  
deseja refazer o teste através de interação com o usuário  
        }  
        #Se a resposta armazenada em respF for sim (s) a função  
"entra" nesse if ,se não for entra no else  
        if(respF=="s"){  
            cat("\tOk, vamos recomeçar o teste.\n")  
            fish.disease()#chamando a função dentro da função para  
recomeçar o teste  
        }else {  
            ###Fim da linha  
            return(cat("\tOk.Que pena!\n\tReavalie os sintomas  
apresentados pelo seu animal \n\tnovamente e, se necessário, refaça o teste.  
\n"))  
        }  
    }  
}  
}  
}  
}  
}else {  
    resp4 <- readline("\tO peixe se mostra barrigudo e com as escamas  
erichadas? \n\tDigite s para sim e n para não: \n")#coletando dado sobre  
sintoma do peixe através de interação com o usuário  
    #enquanto o dado digitado pelo usuário for diferente de s ou n, a  
função deve manter o loop que testa se resp4 é diferente de s ou n e pedir  
que outro caracter seja fornecido  
    while (resp4!="s" & resp4!= "n"){  
        cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para  
não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
```

digitação de respostas válidas para o sintoma.

```
resp4 <- readline("\t0 peixe se mostra barrigudo e com as escamas  
erichadas? \n\tDigite s para sim e n para não: \n")#coletando dado sobre  
sintoma do peixe através de interação com o usuário
```

```
}
```

```
#Se a resposta armazenada em resp(n) for sim (s) a função "entra"  
nesse if ,se não for entra no else
```

```
if(resp4=="s"){
```

```
###Hidropsia, Bacter
```

```
print("Hidropsia")
```

```
#Cálculo da dosagem do tratamento com Bacter: 1 cápsula para cada  
12,5 litros.
```

```
capsulas <- vol/12.5
```

```
####resposta da função a esse conjunto de sintomas:
```

```
return(cat("\tSeu(s) peixe(s) apresenta(m) Hidropsia (ventre  
volumoso). \n\tÉ causada por bactérias que atacam os órgãos internos.  
\n\tOutros sintomas: Manchas vermelhas no corpo, destruição das nadadeiras.  
\n\tTratamento: Adicione ",capsulas," cápsulas(s) de Bacter na água do  
aquário\n"))
```

```
}else {
```

```
resp5 <- readline("\t0 peixe está magro e com o ventre retraído?  
\n\tDigite s para sim e n para não: \n")#coletando dado sobre sintoma do  
peixe através de interação com o usuário
```

```
#enquanto o dado digitado pelo usuário for diferente de s ou n, a  
função deve manter o loop que testa se resp5 é diferente de s ou n e pedir  
que outro caracter seja fornecido
```

```
while (resp5!="s" & resp5!="n"){
```

```
cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para  
não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a  
digitação de respostas válidas para o sintoma.
```

```
resp5 <- readline("\t0 peixe está magro e com o ventre retraído?  
\n\tDigite s para sim e n para não: \n")#coletando dado sobre sintoma do  
peixe através de interação com o usuário
```

```
}
```

```
#Se a resposta armazenada em resp(n) for sim (s) a função "entra"  
nesse if ,se não for entra no else
```

```
if(resp5=="s"){
```

```
###Tuberculose ou Barriga Seca, Bacter
```

```
print("Tuberculose ou Barriga Seca")
```

```
#Cálculo da dosagem do tratamento com Bacter: 1 cápsula para  
cada 12,5 litros.
```

```
capsulas <- vol/12.5
```

```
####resposta da função a esse conjunto de sintomas:
```

```
return(cat("\tSeu(s) peixe(s) apresenta(m) Tuberculose ou  
Barriga Seca. \n\tInfecção bacteriana relacionada com a má qualidade da  
água. \n\tOutros sintomas: Perda de escamas, palidez das cores, destruição  
das nadadeiras. \n\tTratamento: Adicione ",capsulas," cápsulas(s) de Bacter  
na água do aquário.\n"))
```

```
}else {
```

```
resp6 <- readline("\t0 peixe apresenta feridas (buracos) na
```

```

cabeça? \n\tDigite s para sim e n para não: \n")#coletando dado sobre
sintoma do peixe através de interação com o usuário
#enquanto o dado digitado pelo usuário for diferente de s ou n,
a função deve manter o loop que testa se resp6 é diferente de s ou n e
pedir que outro caracter seja fornecido
while (resp6!="s" & resp6!="n"){
  cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para
não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.
  resp6 <- readline("\t0 peixe apresenta feridas (buracos) na
cabeça? \n\tDigite s para sim e n para não: \n")#coletando dado sobre
sintoma do peixe através de interação com o usuário
}
#Se a resposta armazenada em resp(n) for sim (s) a função
"entra" nesse if ,se não for entra no else
if(resp6=="s"){
  ###Buracos na Cabeça, Bacter+Aqualife
  print("Buracos na Cabeça")
  #Cálculo da dosagem do tratamento com Ictio ou Aqualife: 1
gota para 2 litros.
  gotas_Ic_Aq <- vol/2.0
  #Cálculo da dosagem do tratamento com Bacter: 1 cápsula para
cada 12,5 litros.
  capsulas <- vol/12.5
  ####resposta da função a esse conjunto de sintomas:
  return(cat("\tSeu(s) peixe(s) apresenta(m) Buracos na Cabeça
(Hole-in-head). \n\tSintoma ligado ao protozoário Hexamita sp. e
possivelmente a bactérias. \n\tÉ também conhecida como doença dos Ciclídeos.
\n\tOutros sintomas: Inchaço (hidropsia) \n\tTratamento: Adicione
",gotas_Ic_Aq," gota(s) de Aqualife para ",capsulas," cápsulas(s) de Bacter
na água do aquário.\n"))
}else {
  resp7 <- readline("\tAs nadadeiras apresentam-se desfiadas ou
roídas? \n\tDigite s para sim e n para não: \n")#coletando dado sobre
sintoma do peixe através de interação com o usuário
#enquanto o dado digitado pelo usuário for diferente de s ou
n, a função deve manter o loop que testa se resp7 é diferente de s ou n e
pedir que outro caracter seja fornecido
while (resp7!="s" & resp7!="n"){
  cat("\tDigite apenas s minúsculo para sim, ou n minúsculo
para não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.
  resp7 <- readline("\tAs nadadeiras apresentam-se desfiadas
ou roídas? \n\tDigite s para sim e n para não: \n")#coletando dado sobre
sintoma do peixe através de interação com o usuário
}
#Se a resposta armazenada em resp(n) for sim (s) a função
"entra" nesse if ,se não for entra no else
if(resp7=="s"){
  ###Nadadeiras Roídas, Bacter+Aqualife
  print("Nadadeiras Roídas")

```

```
#Cálculo da dosagem do tratamento com Ictio ou Aqualife: 1
gota para 2 litros.
gotas_Ic_Aq <- vol/2.0
#Cálculo da dosagem do tratamento com Bacter: 1 cápsula para
cada 12,5 litros.
capsulas <- vol/12.5
####resposta da função a esse conjunto de sintomas:
return(cat("\tSeu(s) peixe(s) apresenta(m) Nadadeiras
roídas. \n\tCausada por bactérias. Em geral, há ocorrência de fungos
associados. \n\tAs nadadeiras ficam esbranquiçadas e começam a se desfazer.
\n\tTratamento: Adicione ",gotas_Ic_Aq," gota(s) de Aqualife para
",capsulas," cápsulas(s) de Bacter na água do aquário.\n"))
}else {
    resp8 <- readline("\tHá vermelhidão na base das nadadeiras?
\n\tDigite s para sim e n para não: \n")#coletando dado sobre sintoma do
peixe através de interação com o usuário
    #enquanto o dado digitado pelo usuário for diferente de s ou
n, a função deve manter o loop que testa se resp8 é diferente de s ou n e
pedir que outro caracter seja fornecido
    while (resp8!="s" & resp8!="n"){
        cat("\tDigite apenas s minúsculo para sim, ou n minúsculo
para não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a
digitação de respostas válidas para o sintoma.
        resp8 <- readline("\tHá vermelhidão na base das
nadadeiras? \n\tDigite s para sim e n para não: \n")#coletando dado sobre
sintoma do peixe através de interação com o usuário
    }
    #Se a resposta armazenada em resp(n) for sim (s) a função
"entra" nesse if ,se não for entra no else
    if(resp8=="s"){
        ####Septicemia, Bacter+Aqualife
        print("Septicemia")
        #Cálculo da dosagem do tratamento com Ictio ou Aqualife: 1
gota para 2 litros.
        gotas_Ic_Aq <- vol/2.0
        #Cálculo da dosagem do tratamento com Bacter: 1 cápsula
para cada 12,5 litros.
        capsulas <- vol/12.5
        ####resposta da função a esse conjunto de sintomas:
        return(cat("\tSeu(s) peixe(s) apresenta(m) Septicemia.
\n\tResultado de infecção causada por bactérias. \n\tOutros sintomas:
Vasinhos sanguíneos dilatados ao redor dos olhos \n\te na base das
nadadeiras, nadadeiras roídas e desbotadas. \n\tTratamento: Adicione
",gotas_Ic_Aq," gota(s) de Aqualife para ",capsulas," cápsulas(s) de Bacter
na água do aquário.\n"))
    }else {
        #### fim da linha.
        #### Há algumas combinações de sintomas que não levam a
nenhum diagnóstico. Nesse casos, a solução que encontrei foi direcionar o
usuário a refazer
```

[illegible]

```
}  
#Se a resposta armazenada em resp(n) for sim (s) a função "entra" nesse  
if ,se não for entra no else  
if(resp2=="s"){  
  resp3 <- readline("\tAs escamas estão eriçadas e as nadadeiras  
fechadas? \n\tDigite s para sim e n para não: \n")#coletando dado sobre  
sintoma do peixe através de interação com o usuário  
  #enquanto o dado digitado pelo usuário for diferente de s ou n, a  
função deve manter o loop que testa se resp3 é diferente de s ou n e pedir  
que outro caracter seja fornecido  
  while (resp3!="s" & resp3!= "n"){  
    cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para não.  
\n\tApós digitar, aperte enter.\n") #Alertando o usuário para a digitação de  
respostas válidas para o sintoma.  
    resp3 <- readline("\tHá um limo branco acizentado, cobrindo as  
brânquias e grande extensão do corpo? \n\tDigite s para sim e n para não:  
\n")#coletando dado sobre sintoma do peixe através de interação com o  
usuário  
  }  
  #Se a resposta armazenada em resp(n) for sim (s) a função "entra"  
nesse if ,se não for entra no else  
  if(resp3=="s"){  
    ###Água muito ácida, Alkali  
    print("Água Muito Ácida")  
    #Cálculo da dosagem do tratamento com Alkali ou Acid: 1 gota para 3  
litros.  
    gotas_Al_Ac <- vol/3.0  
    ####resposta da função a esse conjunto de sintomas:  
    return(cat("\tA água do seu aquário deve estar muito ácida.  
\n\tOutros sintomas: Nadadeiras roídas e desbotadas. \n\tTratamento:  
Verifique as condições de pH. \n\tConfirmando a acidez excessiva, adicione  
",gotas_Al_Ac," gota(s) de Acid na água do aquário. \n"))  
  }else {  
    resp4 <- readline("\tO peixe apresenta respiração ofegante próximo à  
superfície e perda de brilho das escamas? \n\tDigite s para sim e n para  
não: \n")#coletando dado sobre sintoma do peixe através de interação com o  
usuário  
    #enquanto o dado digitado pelo usuário for diferente de s ou n, a  
função deve manter o loop que testa se resp4 é diferente de s ou n e pedir  
que outro caracter seja fornecido  
    while (resp4!="s" & resp4!= "n"){  
      cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para  
não. \n\tApós digitar, aperte enter.\n") #Alertando o usuário para a  
digitação de respostas válidas para o sintoma.  
      resp4 <- readline("\tO peixe apresenta respiração ofegante próximo  
à superfície e perda de brilho das escamas? \n\tDigite s para sim e n para  
não: \n")#coletando dado sobre sintoma do peixe através de interação com o  
usuário  
    }  
    #Se a resposta armazenada em resp(n) for sim (s) a função "entra"
```


Há algumas combinações de sintomas que não levam a nenhum diagnóstico. Nesse casos, a solução que encontrei foi direcionar o usuário a refazer

o teste de modo a repensar se os sintomas que ele apontou anteriormente estavam certos:

```
respF <- readline("\tNão foi possível identificar a doença que acomete  
seu peixe por meio dos sintomas informados.\n\tVamos refazer o teste?  
\n\tDigite s para sim e n para não: \n")#Verificando se o usuário deseja  
refazer o teste através de interação com o usuário
```

#enquanto o dado digitado pelo usuário for diferente de s ou n, a função deve manter o loop que testa se respF é diferente de s ou n e pedir que outro caracter seja fornecido

```
while (respF!="s" & respF!= "n"){  
  cat("\tDigite apenas s minúsculo para sim, ou n minúsculo para não.  
\n\tApós digitar, aperte enter.\n") #Alertando o usuário para a digitação de  
respostas válidas para o respF.
```

```
  respF <- readline("\tNão foi possível identificar a doença que  
acomete seu peixe por meio dos sintomas informados.\n\tVamos refazer o  
teste? \n\tDigite s para sim e n para não: \n")#Verificando se o usuário  
deseja refazer o teste através de interação com o usuário
```

```
  }  
  #Se a resposta armazenada em respF for sim (s) a função "entra" nesse  
if ,se não for entra no else
```

```
  if(respF=="s"){  
    cat("\tOk, vamos recomeçar o teste.\n")  
    fish.disease()#chamando a função dentro da função para recomeçar o
```

teste

```
  }else {
```

###Fim da linha

```
    return(cat("\tOk. Que pena!\n\tReavalie os sintomas apresentados pelo  
seu animal \n\tnovamente e, se necessário, refaça o teste. \n"))
```

```
  }
```

```
}
```

```
}
```

```
}
```

fish.disease()

From:

<http://ecor.ib.usp.br/> - ecoR

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2019:alunos:trabalho_final:camila.cristina.martins:funcao

Last update: 2020/08/12 06:04