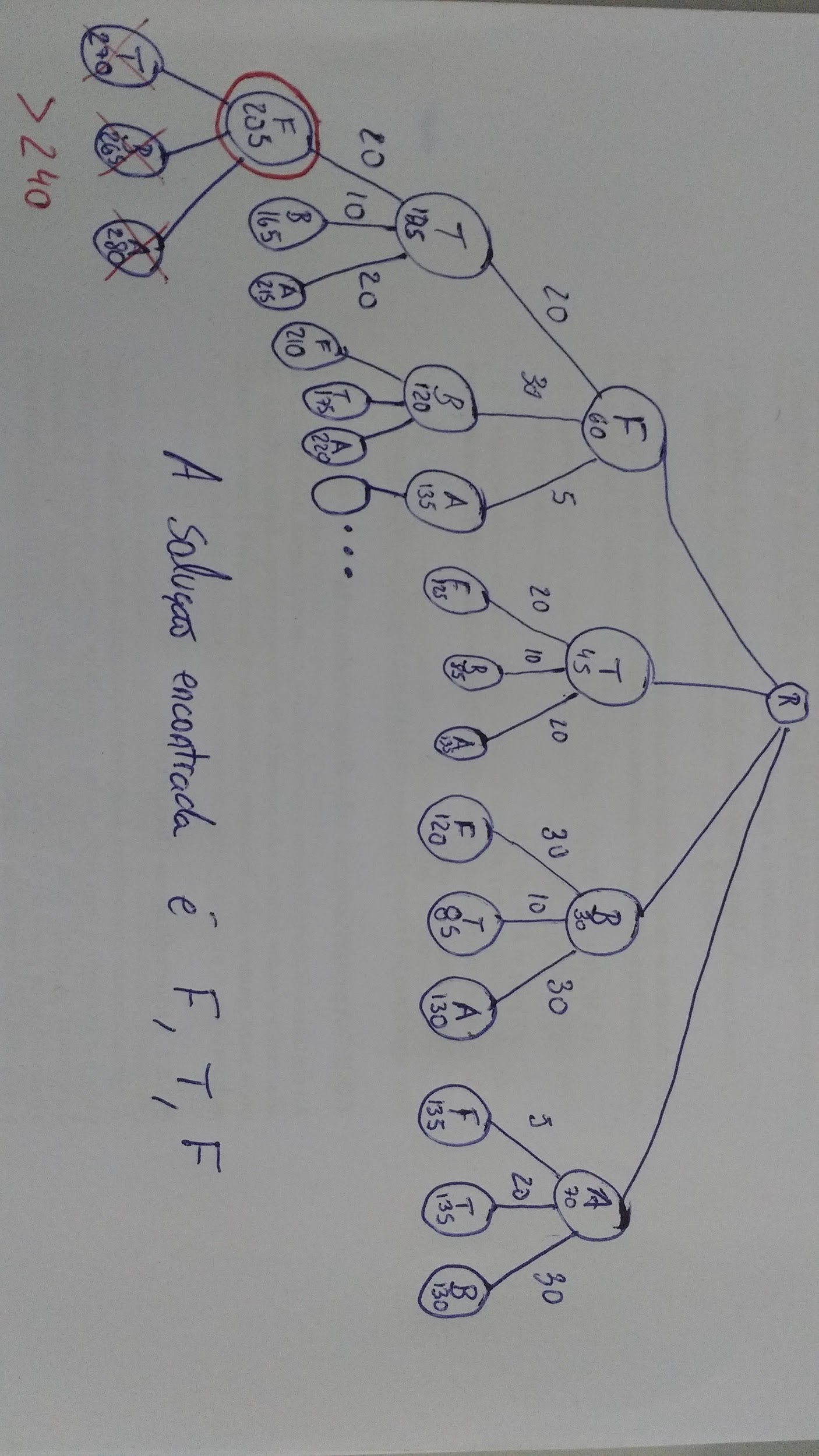
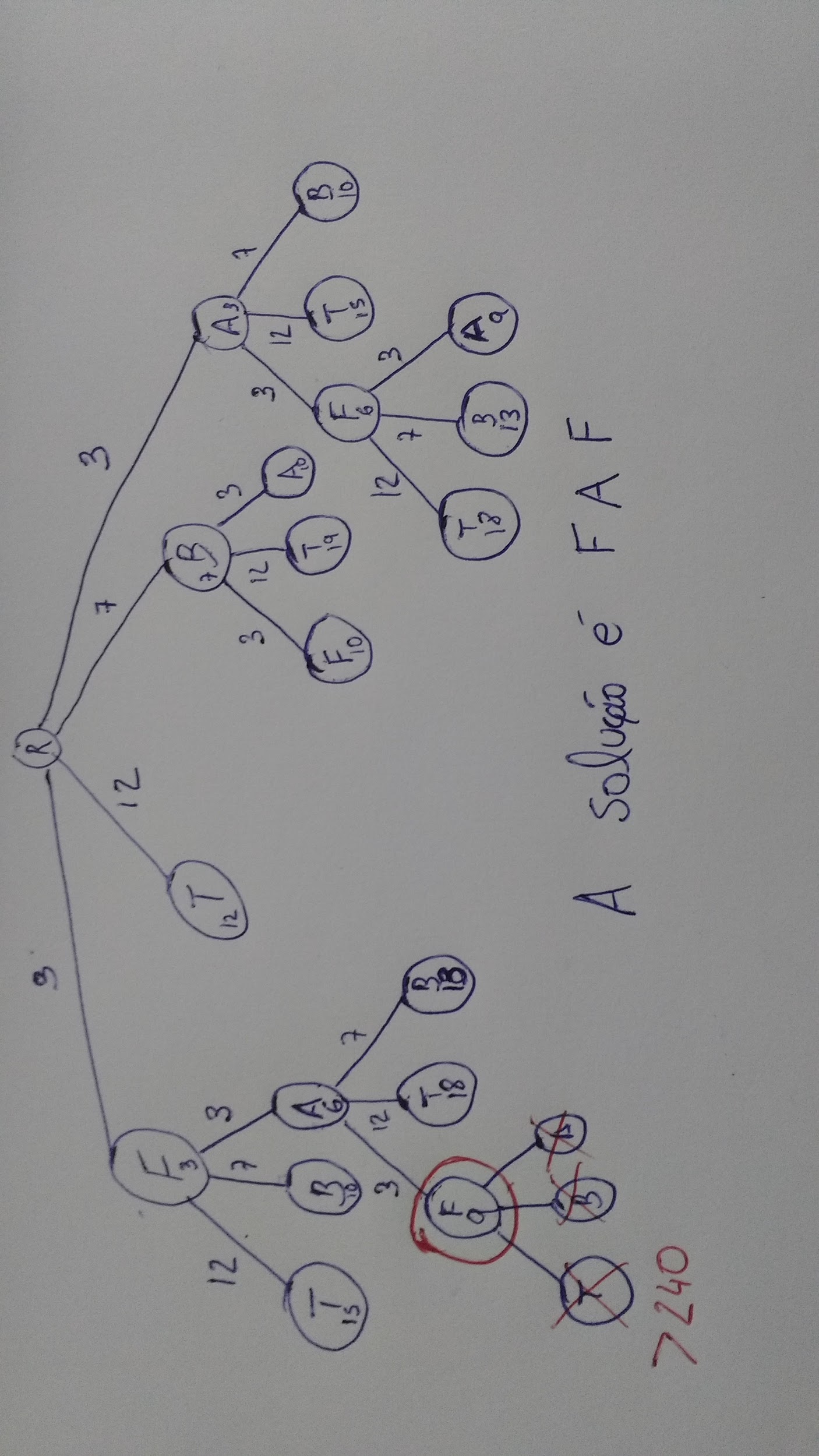
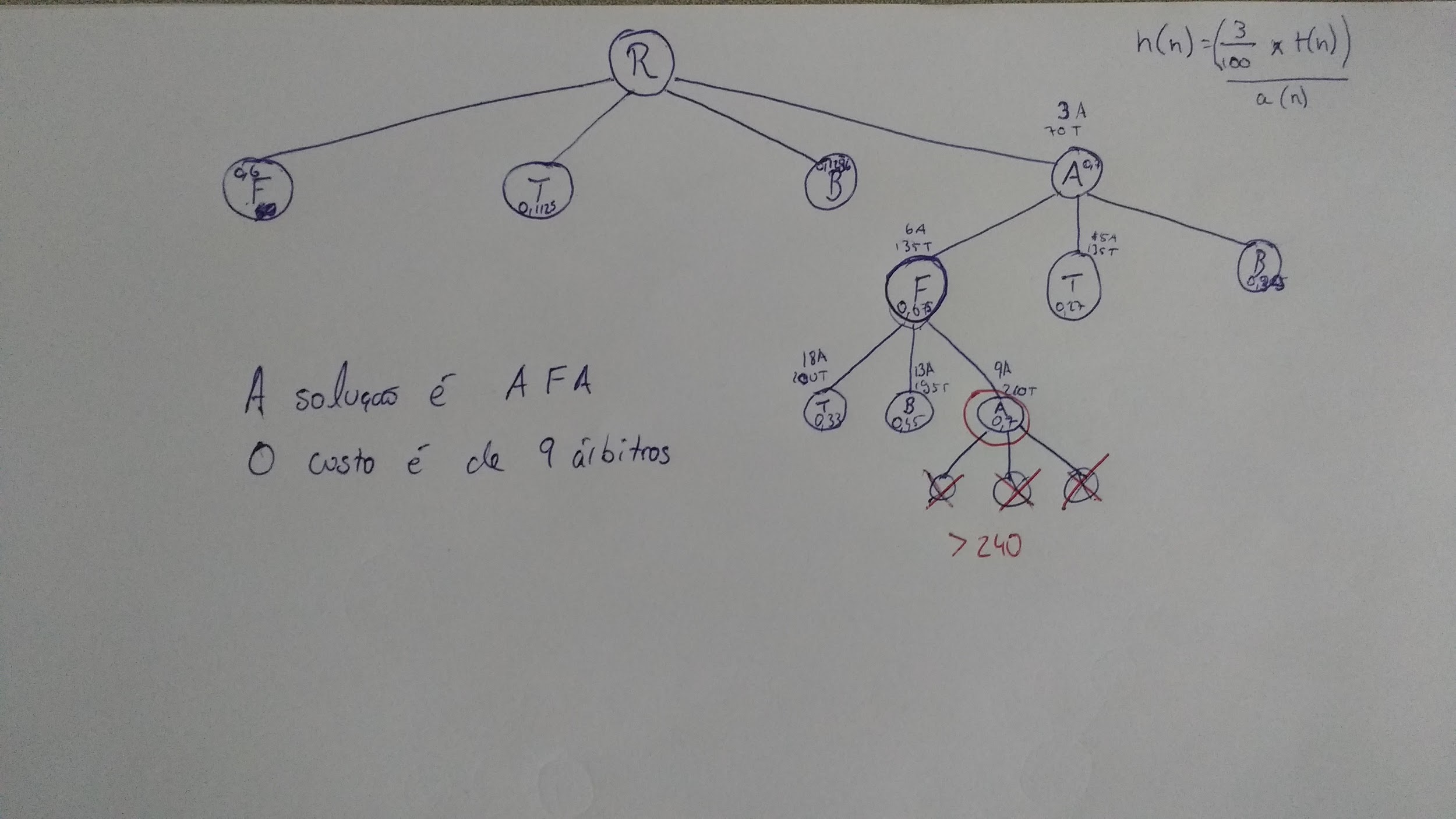
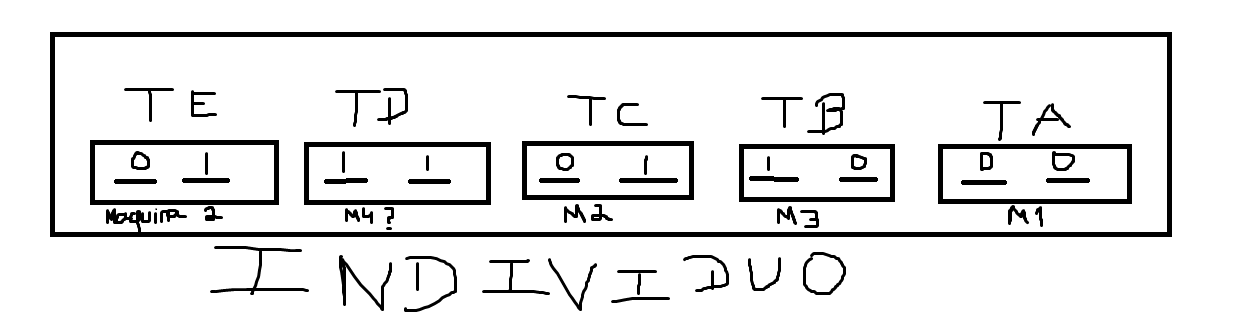
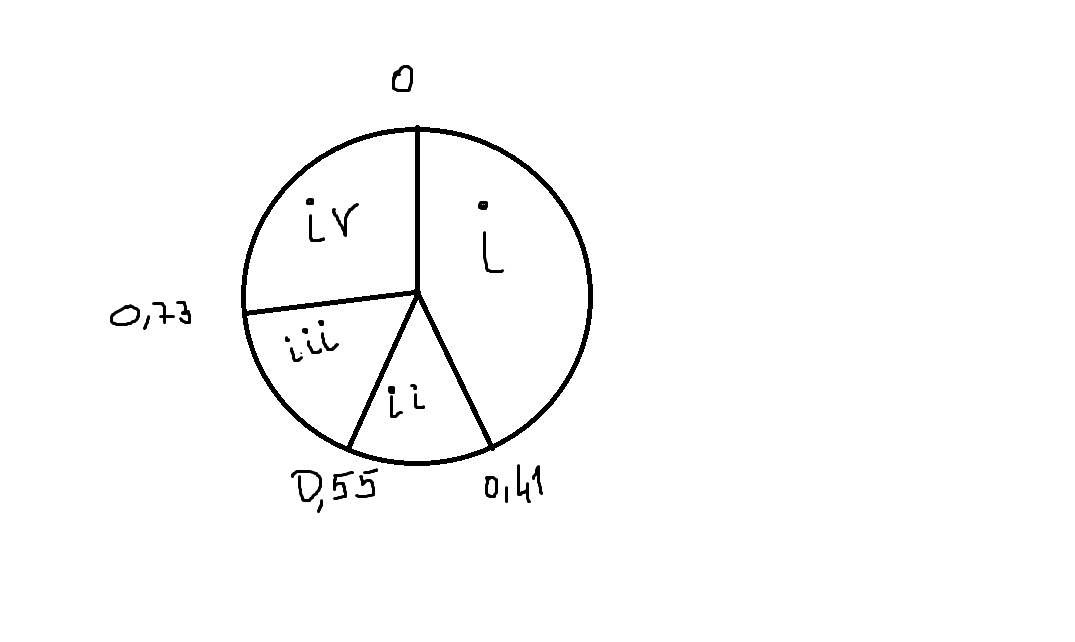
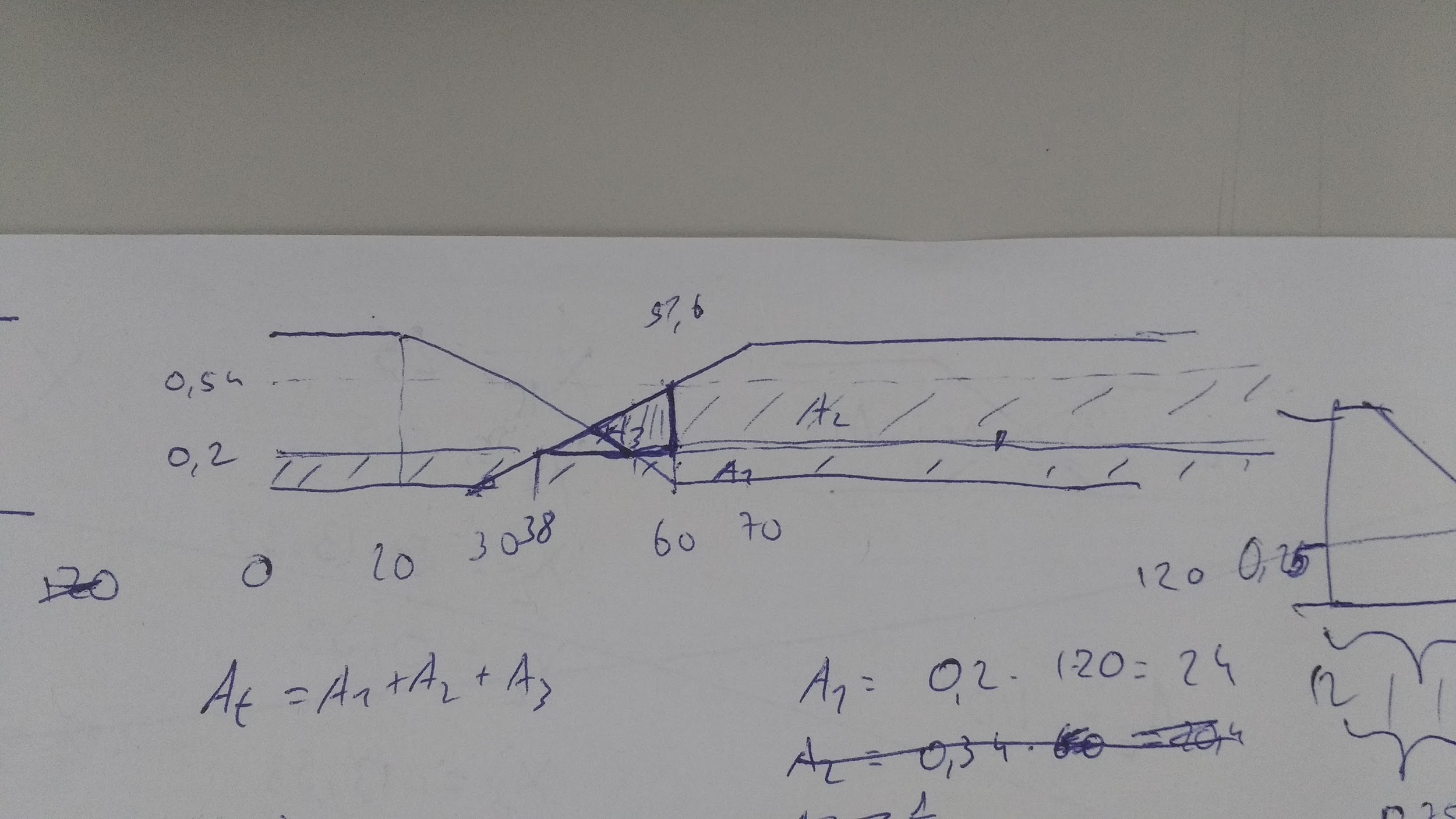
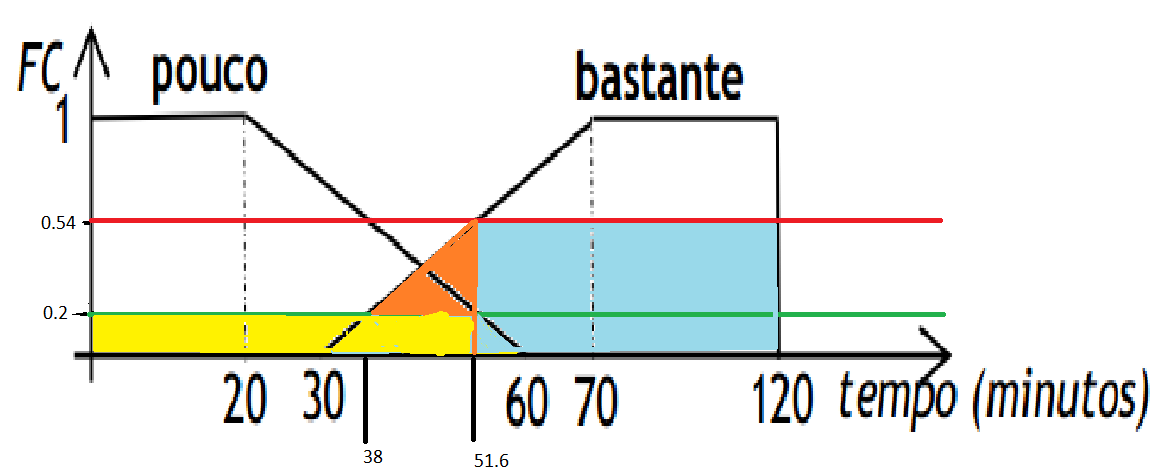
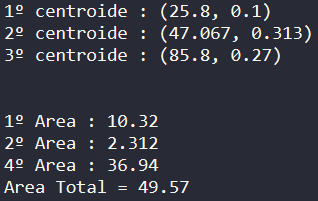
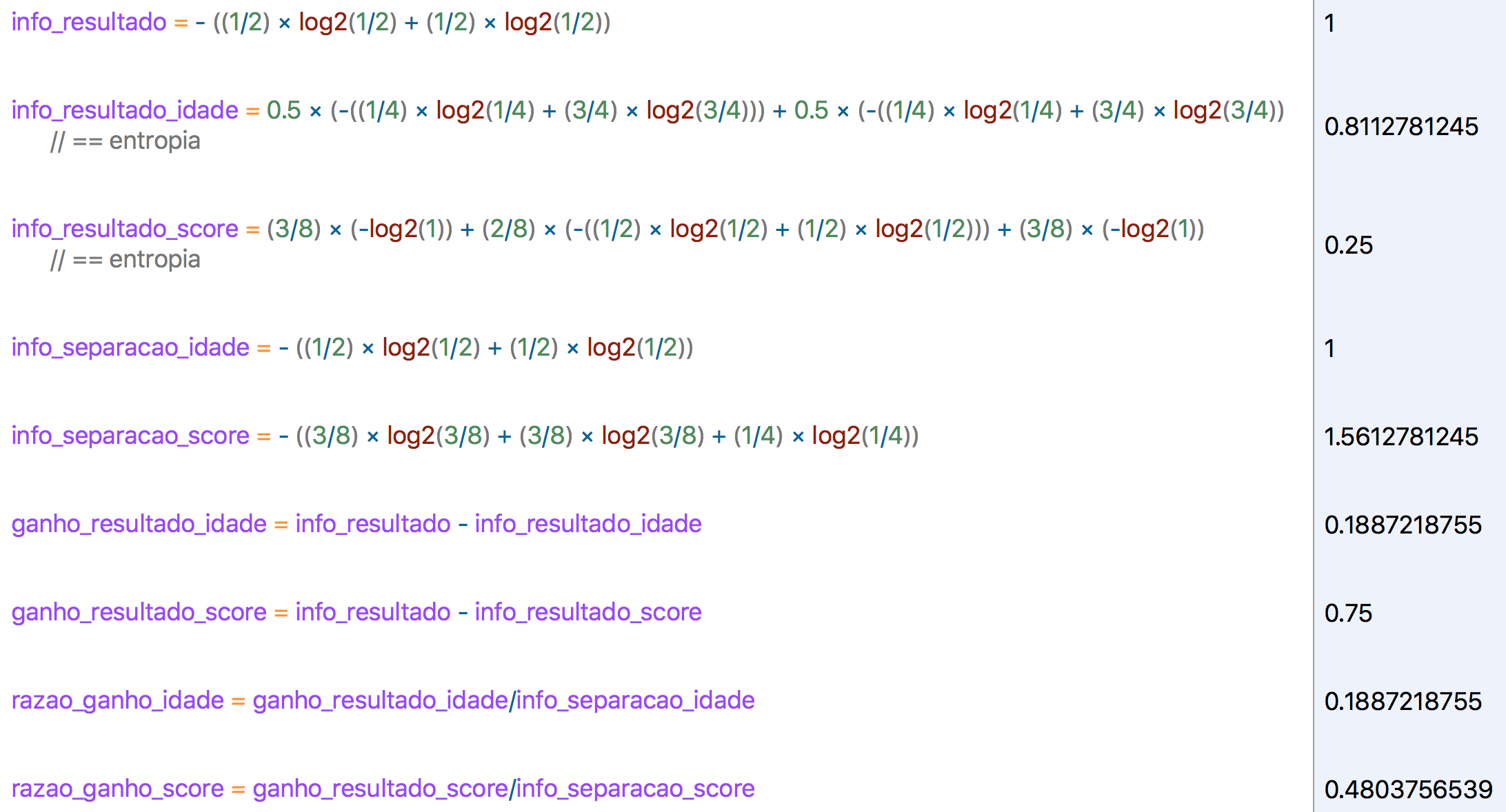
Resolução Exame 2012-2013 Normal

* 1. 
  2. O custo da solução é de 9 árbitros.
  3. A heurística admissível sugerida é a seguinte:  
       
     h+(n)=custo(n)/tempo\_max(n)\*(240-tempo\_acumulado(n)) = 3/100 \* (240 - tempo\_acumulado(n)) 
  4. 
  5. Representação:  
       
     Escolhi esta representação porque é intuitiva, simples e além disso a probabilidade de um indivíduo sobreviver é grande.  
     A probabilidade de um indivíduo sobreviver é de 0,75^5 = 0.237.  
     O individuo é composto por uma cadeia de 10 bits em que cada 2 bits representando uma máquina da tarefa.
     1. 10 01 10 00 01
     2. 10 01 10 10 10
     3. 01 01 00 00 00
     4. 01 01 10 00 01
     5. aval = 3 \* (5 / 10 + 10 / 10 + 10 / 10) = 3 \* (0.5 + 1 + 1) = 3 \* 2.5 = 7.5
     6. aval = 2 \* (0 / 20 + 5 / 20 + 20 / 20) = 2 \* (0 + 0.25 + 1) = 2 \* 1.25 = 2.5
     7. aval = 2 \* (15 / 15 + 10 / 15 + 0 / 15) = 2 \* (1 + 0.67 + 0) = 2 \* 1.67 = 3.34
     8. aval = 3 \* (5 / 15 + 15 / 15 + 5 / 15) = 3 \* (0.33 + 1 + 0.33) = 3 \* 1.66 = 5
  6. Como existe elitismo para o melhor o indivíduo i é automaticamente selecionado.  
       
     Calcular probabilidades relativas dos restantes indivíduos  
       
     ∑ f(xi) = 5 + 3.34 + 2.5 = 18.33  
       
     p(i) = 7.5 / 18.33 = 0.41  
     p(ii) = 2.5 / 18.33 = 0.14  
     p(iii) = 3.34 / 18.33 = 0.18  
     p(iv) = 5 / 18.33 = 0.27  
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
     Tendo em conta a geração dos números aleatórios os indivíduos escolhidos foram:  
      - iv  
      - i  
      - ii  
       
     Ficamos então com os indivíduos i, iv, i, ii.  
     
  7. Para cruzar os indivíduos iremos usar o emparelhamento aleatório.  
       
     O melhor (i) passa para a geração seguinte.  
     O indivíduo iv não cruza uma vez que 0.75 < 0.88 sendo assim passa também para a geração seguinte.  
       
     Os indivíduos que irão cruzar são o i e o ii usando a estratégia “One-point crossover” sendo este ponto aleatório mas múltiplo de 2. Desta forma garantimos uma maior variedade de indivíduos na próxima geração com uma maior probabilidade de sobreviverem uma vez que sem mutação herda tarefas que já são válidas.   
       
     Cruzamento:  
     10 01 10 | 00 01 => 10 01 10 10 01  
     10 01 10 | 10 01 => 10 01 10 00 01 => Mutação => 10 00 10 00 01  
       
     Nova geração:  
     i - 10 01 10 00 01  
     ii - 10 01 10 10 01  
     iii - 10 00 10 00 01  
     iv - 01 01 10 00 01
  8. MB(collie) = 1.0  
     MB(ter\_dono) = 1.0  
     MB(calor) = 0.6  
     MB(frio) = 0.25  
     MB(~chover) = 1.0  
     MB(treinado) = 0.8  
     MB’(muito\_tempo, calor ∩ ~chover) = 0.9  
     MB’(pouco\_tempo, frio ∪ chover) = 0.8  
     MB’(bom\_cao, spaniel ∪ collie) = 0.7  
     MB’(bom\_cao, treinado ∩ ter\_dono) = 0.9  
     MB’(com\_dono, bom\_cao) = 0.9  
     \  
     MB(calor ∩ ~chover) = min(MB(calor), MB(~chover)) = min(0.6, 1) = 0.6  
     MB(frio ∪ chover) = max(MB(frio), MB(chover)) = max(0.25, 0) = 0.25  
     MB(spaniel ∪ collie) = max(MB(spaniel), MB(collie)) = max(0, 1.0) = 1.0  
     MB(treinado ∩ ter\_dono) = min(MB(treinado), MB(ter\_dono)) = min(0.8, 1.0) = 0.8  
       
     MB(muito\_tempo, calor ∩ ~chover) = MB’(muito\_tempo, calor ∩ ~chover) \* MB(calor ∩ ~chover) = 0.9 \* 0.6 = 0.54  
     MB(pouco\_tempo, frio ∪ chover) = MB’(pouco\_tempo, frio ∪ chover) \* MB(frio ∪ chover) = 0.8 \* 0.25 = 0.2  
     MB(bom\_cao, spaniel ∪ collie) = MB’(bom\_cao, spaniel ∪ collie) \* MB(spaniel ∪ collie) = 0.7 \* 1.0 = 0.7  
     MB(bom\_cao, treinado ∩ ter\_dono) = MB’(bom\_cao, treinado ∩ ter\_dono) \* MB(treinado ∩ ter\_dono) = 0.9 \* 0.8 = 0.72  
       
     MB(bom\_cao) = MB(bom\_cao, spaniel ∪ collie) + (1 - MB(bom\_cao, spaniel ∪ collie)) \* MB(bom\_cao, treinado ∩ ter\_dono) = 0.7 + 0.3 \* 0.72 = 0.916  
       
     MB(com\_dono, bom\_cao) = MB’(com\_dono, bom\_cao) \* MB(bom\_cao) = 0.9 \* 0.916 = 0.824  
       
     MB(muito\_tempo ∩ com\_dono) = min(MB(muito\_tempo), MB(com\_dono)) = min(0.54, 0.824) = 0.54  
     MB(pouco\_tempo ∩ com\_dono) = min(MB(pouco\_tempo), MB(com\_dono)) = min(0.2, 0.824) = 0.2  
     FC(muito\_tempo ∩ com\_dono) = MB(muito\_tempo ∩ com\_dono) = 0.54  
       
     Pode-se concluir com fator de certeza de 0.54 que o Bobby está no parque muito tempo.
  9. O Bobby passará 45.308 minutos no parque.  
       
       
       
       
       
       
       
       
       
     A1 = 0.2 \* 120 = 24  
     A2 = 0.34 \* (120 - 51.6) = 0.34 \* 68.4 = 23.256  
     A3 = 0.34 \* 1 / 2 \* (51.6 - 38) = 2.312  
       
     At = A1 + A2 + A3 = 49.568

X1 = 120 / 2 = 60  
 X2 = 68.4 / 2 + 60 = 34.2 + 60 = 94.2  
 X3 = 1 / 3 \* 13.6 + 38 = 4.533 + 38 = 42.53  
  
 Xc = ((X1 \* A1) + (X2 \* A2) + (X3 \* A3)) / At = 75.230



Xc = 71.51

* 1. Falso. Pois o facto de heurística ser admissível não garante que a solução seja ótima pelo que o resultado entre os dois algoritmos poderá ser diferente.
  2. Os algoritmos de pesquisa local são caraterizado por usar uma função heurística que permite dar valor a um determinado estado para desta forma se escolher aquela que se acredita ser a melhor opção, ignorando as outras. A grande diferença entre o “subir a colina” e o “arrefecimento simulado” é que enquanto o “subir a colina” apenas aceita estados melhores que o atual, o “arrefecimento simulado” tolera (com uma probablidade) a aceitação de um estado pior que o atual. Desta forma evita ficar preso em máximos e mínimos locais o que acontece no “subir a colina”.
  3. Ao ordenar os nós gerados fica possível ao algoritmo aplicar um corte alpha-beta mais cedo uma vez que é possível detetar piores jogadas mais cedo do que se não estiverem ordenados.
  4. Ganho de informação: o melhor atributo é o score uma vez que é aquele com menor entropia; veja-se que com o score igual a 3 é sabido que todos têm um resultado “Sim” e com score igual a 1 todos têm um resultado “Não”.  
     Tanto usando o critério razão de ganho como o ganho de informação, o score é o melhor atributo. 
  5. A aprendizagem simbólica, também conhecida por aquisição de conhecimentos é adquirido através da aquisição de novas informações sendo capaz de aplicar o novo conhecimento em situações novas. Por exemplo “Aprender inteligência artificial por um livro”. Já a aprendizagem não simbólica, conhecida por refinamento de habilidade consiste no processo de refinamento de uma habilidade por sucessivas tentativas como por exemplo “andar de bicicleta”.
  6. verbo(subir) → [subiu].  
     verbo(descer) → [desceu].  
     nome → [\_].  
     numero(N) → [N], { number(N) }.  
     valor(1) → [valor].  
     valor(\_) → [valores].  
       
     verifica(Acao, Numero1, Numero2) :-   
      (Acao → subir, Numero1 < Numero2; Numero1 > Numero2).  
       
     frase(Acao, Numero1, Numero2) → nome, sintagma\_verbal(Acao, Numero1, Numero2), !, {verifica(Acao, Numero1, Numero2)}.  
     sintagma\_verbal(Acao, Numero1, Numero2) →   
      verbo(Acao),  
      [de],  
      numero(Numero1),  
      valor(Numero1),  
      numero(Numero2),  
      [para],  
      numero(Numero2),  
      valor(Numero2).
  7. Overfitting consiste em treinar a rede neuronal em demasia, de forma a que esta se adeque demasiado bem ao *training set*. Desta forma torna-se incapaz de analisar corretamente novas entradas e atribui-lhes valores semelhantes aos de treino. Para evitar isto, deve-se garantir que a rede tem uma dimensão grande o suficiente para não se adaptar inteiramente aos dados de treino e também utilizar dados de teste diferentes dos de treino.