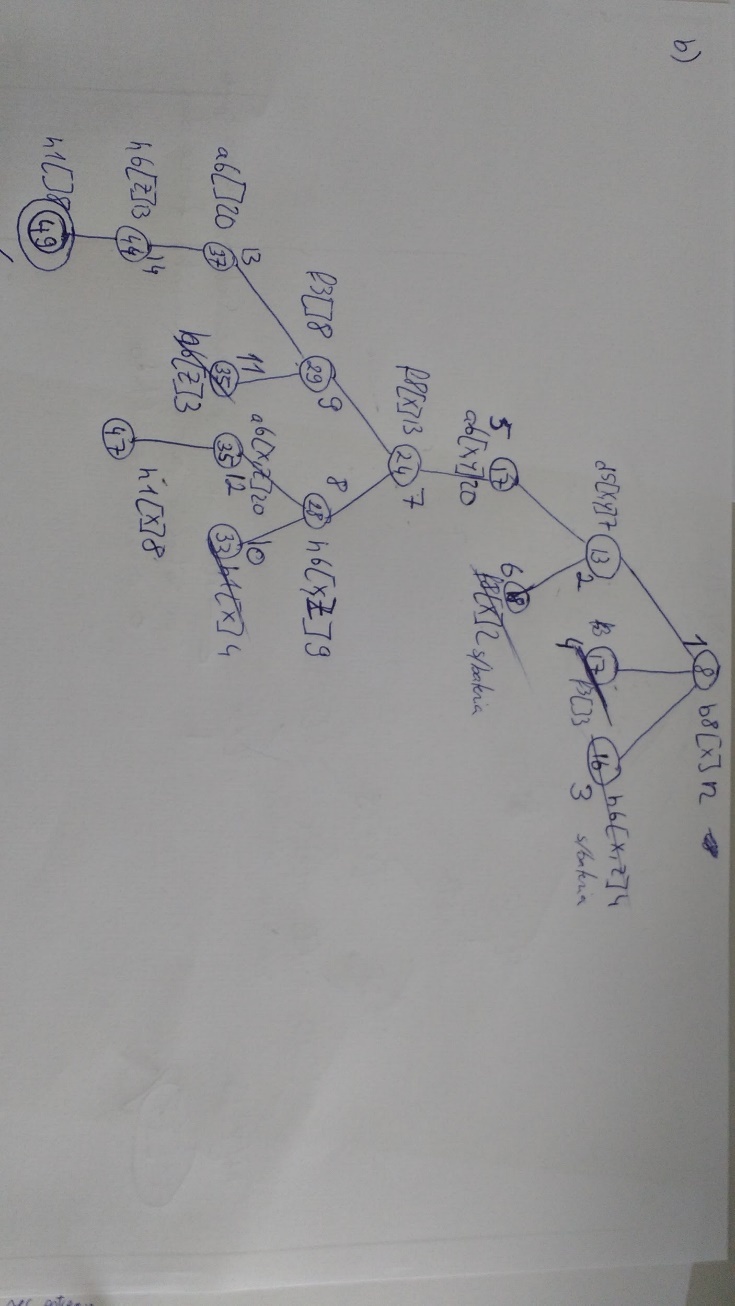
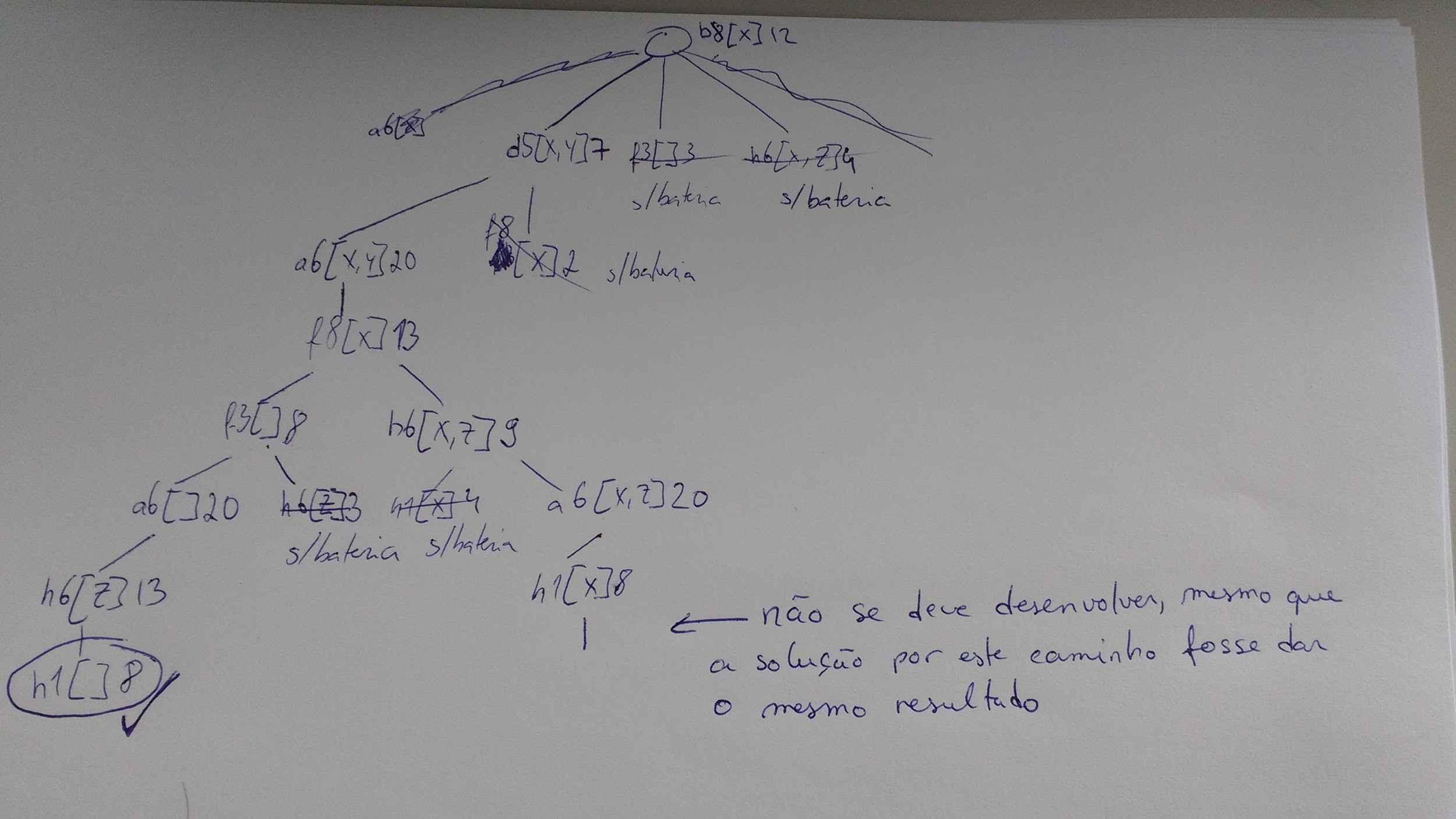
Resolução Exame 2013-2014 Normal

1. 

   2. h\*(n) = número de peças por entregar



1. a) A quantidade max = 6 -> 3 bits | B qtd máx = 2 -> 2 bits | C qtd max = 3 -> 2bits

i) 011 01 01 -> A(011) B(01) C(01)

ii) 101 01 10

iii) 001 10 11

iv) 101 10 00

qa= quantidade de a | pa= peso unitario de a | va = valor de a

funcao de adaptacao = (qa\*pa + qb\*pb + qc\*pc) / 300 -> se funcao der maior que 1, descarta ( esta deve ser suficiente uma vez que quanto mais peso, mais valor…)

/\* fadaptacao2 = (qa\*va +qb\*vb + qc\*vc) / (6\*20+2\*30+3\*40) quanto maior esta melhor não entra em conta com o peso :(

funcao de adaptacao com tudo ao barulho = (qa\*(va/pa) + qb\*(vb/pb) + qc\*(vc/pc)) / …. por completar \* /

fa(i) = (3\*30 + 50 + 60) / 300 = 2/3

fa(ii) = 320/300 = 1.06 -> excede o peso

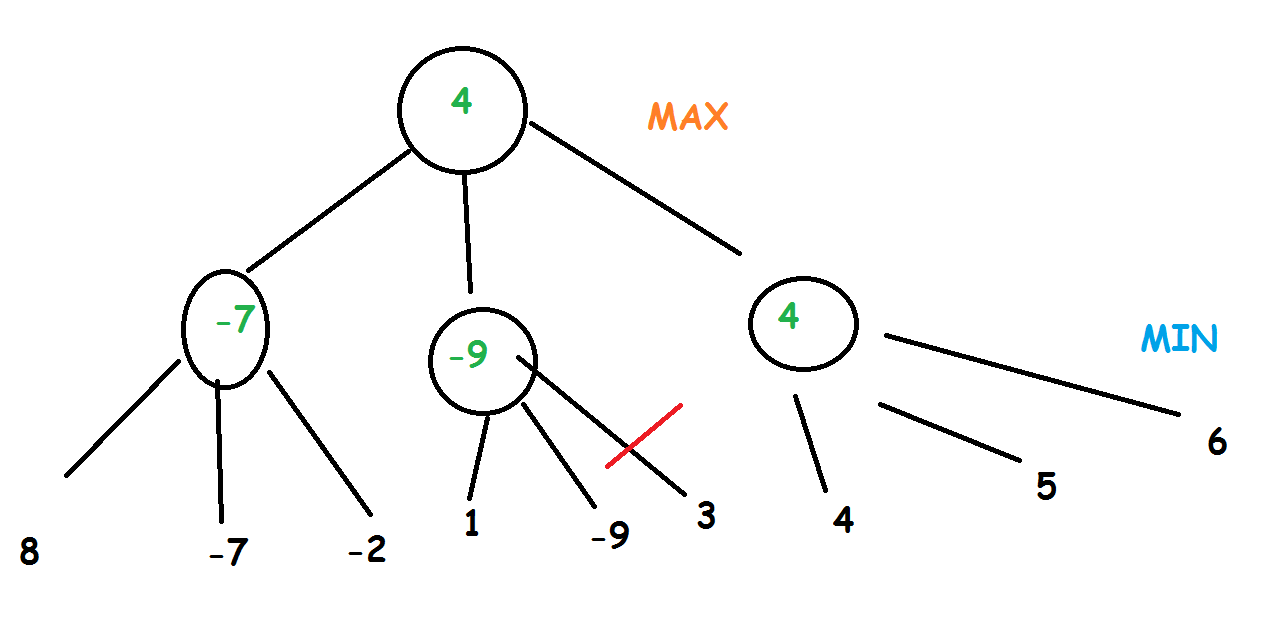
fa(iii) = 310/300 -> excede o peso

fa(iv) = 250/300 -> 0,833

b) individuo iv -> elitista

b)

% Factos  
ganhar(alemanha, portugal).  
correr(ronaldo, pouco).  
correr(pepe, pouco).  
correr\_relativo(mais, ronaldo, pepe).  
correr\_relativo(menos, pepe, ronaldo).  
  
% Gramatica  
equipa(portugal) --> ['Portugal'].  
equipa(alemanha) --> ['Alemanha'].  
vvb  
jogador(ronaldo) --> ['Ronaldo'].  
jogador(pepe) --> ['Pepe'].  
  
verbo(ganhar) --> [ganhou].  
verbo(correr) --> [correu].  
  
adverbio(muito) --> [muito].  
adverbio(pouco) --> [pouco].  
  
adverbio(mais) --> [mais], [do], [que].  
adverbio(menos) --> [menos], [do], [que].  
  
preposicao --> ['à'];[a].  
  
% Sintaxe  
frase(A, S, O) -->  
 sintagma\_nominal(S),  
 sintagma\_verbal(A, S, O).  
  
sintagma\_nominal(S) -->  
 equipa(S).  
sintagma\_nominal(S) -->  
 jogador(S).  
  
% Ganhar  
sintagma\_verbal(A, S, O) -->  
 verbo(A),  
 preposicao,  
 { jogador(O) ; equipa(S) },  
 { write('Erro Semantico') }.  
sintagma\_verbal(A, S, O) -->  
 verbo(A),  
 preposicao,  
 equipa(O),  
 { (ganhar(S, O), write('Concordo'); write('Discordo')) }.  
  
sintagma\_verbal(A, S, O) -->  
 verbo(A),  
 adverbio(QT),  
 { correr(S, QT), write('Concordo'); write('Discordo') }.  
  
sintagma\_verbal(A, S, O) -->  
 verbo(A),  
 adverbio(QT),  
 jogador(O),  
 { correr\_relativo(QT, S, O), write('Concordo'); write('Discordo') }.  
sintagma\_verbal(A, S, O) -->  
 verbo(A),  
 adverbio(QT),  
 { jogador(S) ; equipa(O) },  
 { write('Erro Semantico') }.

* 1. Depende. Para h € ]0 ; 1[ -> h^2 poderá não ser admissível, já sqrt(h) irá continuar a ser. Para valores de h >= 1 h^2 irá continuar a ser admissível, já sqrt(h) poderá não o ser. Em ambos os casos a árvore de pesquisa iria ser rigorosamente a mesma.
  2. Escolheria o algoritmo de “custo uniforme” uma vez que a complexidade temporal é menor do que as alternativas e também garante uma solução ótima.
  3. Regra de diagnóstico: Máquina Alvo i ∩ Xi > Xr -> Virado Direita (FC=(Xi - Xr) / 100). Tendo em conta que a máquina alvo é a “i” e a posição x do robot é inferior à da máquina existe um fator de certeza de que está virado para a direita que é dependente da distância atual à máquina.  
     Regra causal: Máquina Alvo i ∩ Xi > Xr ∩ Nao Virado Direita-> Rodar Direita. Se a máquina alvo “i” estiver à direita do robot e ele não estiver virado para a direita então rodar o robot para a direita.
  4. 
  5. info(C) = - E p(ci) \* log2(p(ci)) = -(0.5\*log2(0.5) + 0.5\*log(0.5)) = 1  
     info(C|A1) = 1/3 \* (-1/2 \* log2(1/2) - 1/2 \* log2(1/2)) + 1/3 \* (-1/2 \* log2(1/2) - 1/2 \* log2(1/2)) + 1/3 \* (-1/2 \* log2(1/2) - 1/2 \* log2(1/2))+  
     G(C|A1) = FC(A) \* (info(C) - info(C|A1)) = 1 \* (1 - 1) = 0  
     InfoSeparacao(A1) = 1/3 \* log2(1/3) \* 3 = 1.585  
     RG(C|A1) = G(C|A1) / InfoSeparacao(A1) = 0 / 1.585 = 0
  6. Iria utilizar a aprendizagem de Hopfield.
  7. Plausibilidade(CM) = 1 - Crença(~CM)  
      = 1 - MB(~CM)  
      = 1 - 0.8  
      = 0.2