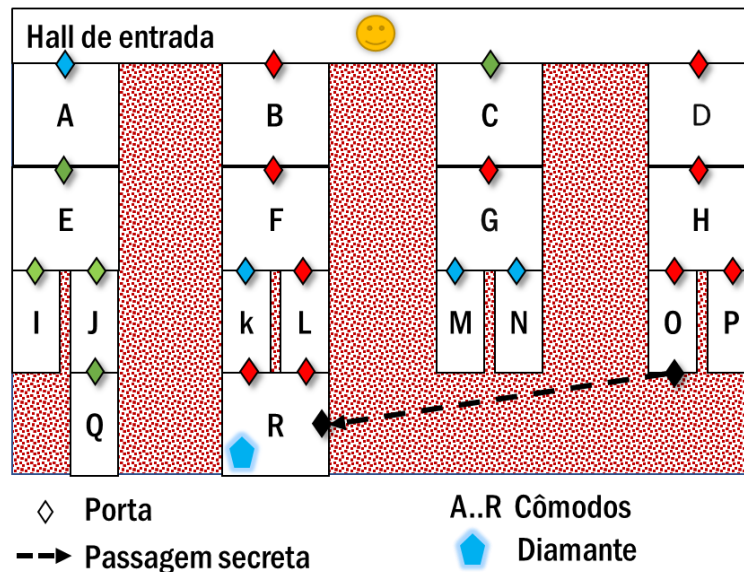


## UTFPR-CURITIBA – EXERCÍCIO BUSCA CEGA E CUSTO-UNIFORME

**Problema:** O agente deve encontrar um caminho para recuperar o diamante azul que foi roubado. No entanto, só pode passar de um cômodo para outro desde que estejam ligados por uma porta ou por uma passagem secreta. O agente possui conhecimento prévio do mapa do castelo que lhe foi passado por um espião. Inicialmente ele está no Hall de Entrada. O agente sempre escolhe a passagem por ordem alfabética em caso de empate. Losangos representam portas ou passagens. O **custo** de movimentação varia de acordo com a cor do losango. Losangos **vermelhos** têm custo 5, **azul** 3, **verde** 1 e **preta** 0,5. As passagens são bidirecionais.



**Objetivo:** comparar busca em profundidade, largura (extensão) e de custo uniforme (cheapest-first) para o problema de encontrar um caminho do hall de entrada até a posição do diamante.

**FAZER:**

- 1) Desenhar a árvore de busca e apresentar a solução encontrada e o custo da mesma por estratégia:
  - a. Profundidade sem recursividade;
  - b. Profundidade com recursividade;
  - c. Largura; e
  - d. Custo-uniforme.
- 2) Para o cenário apresentado, comparar as estratégias de busca em termos de:
  - a. Complexidade temporal: número de nós gerados – conte também os nós que foram gerados temporariamente e não foram incluídos na fronteira (porque já havia um melhor);
  - b. Complexidade espacial: número máximo de nós que tiveram que ser mantidos em memória considerando-se cada *loop* de execução (complexidade espacial),
  - c. Otimalidade: se consegue encontrar a solução de menor custo e
  - d. Compleitude: se consegue encontrar a solução.

## RESPOSTAS

1.

	Prof (sem recurs.)	Prof. Recur	Largura	Custo unif.
Nós gerados	20 (13)	17 (10)	30 (19)	38 (21)
Nós em memória	8	5	30 (19)	38 (21)
Custo	18	18	18	15,5
Otimalidade	Não	Não	Não	Sim
completude	Sim	Sim	Sim	Sim
Solução	BFKR	BFKR	BFKR	DHOR

Valores entre parênteses foram obtidas considerando-se que as passagens só possuem uma direção.

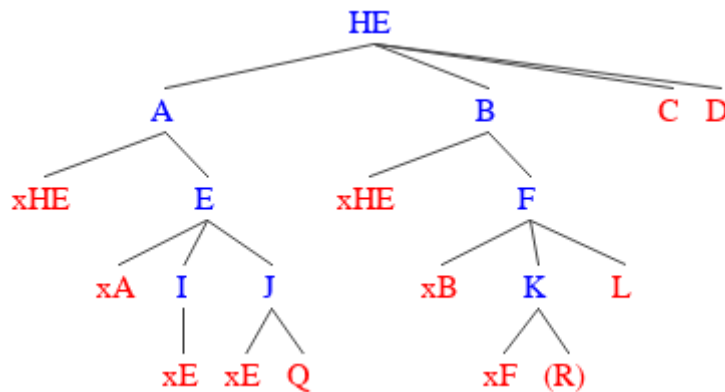
### PROFUNDIDADE SEM RECURSIVIDADE

#### Legenda

x<Nó>: nó não incluído na fronteira porque <Nó>.estado já foi explorado

-<Nó>: nó não incluído na fronteira porque <Nó>.estado já está na fronteira e é melhor

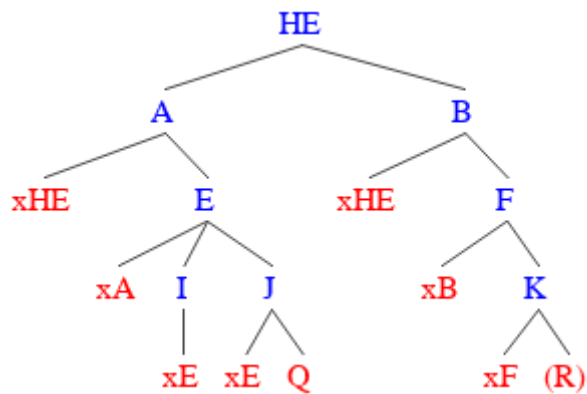
(<Nó>): nó que representa a solução



<http://ironcreek.net/phpsyntaxtree/?> [HE [A [xHE] [E [xA][I[xE]][J[xE][Q]]]] [B[xHE][F[xB][K[xF][(R)]]][L]]] [C] [D]]

- Nós gerados: 20 (se não contar os já explorados x<Nó> = 13)
- Máximo de nós em memória: 8 (ao explorar o caminho HAEJQ estão na memória os nós HABCDEJQ)
- Solução encontrada:  $ir(B) > ir(F) > ir(K) > ir(R)$
- Custo = 18

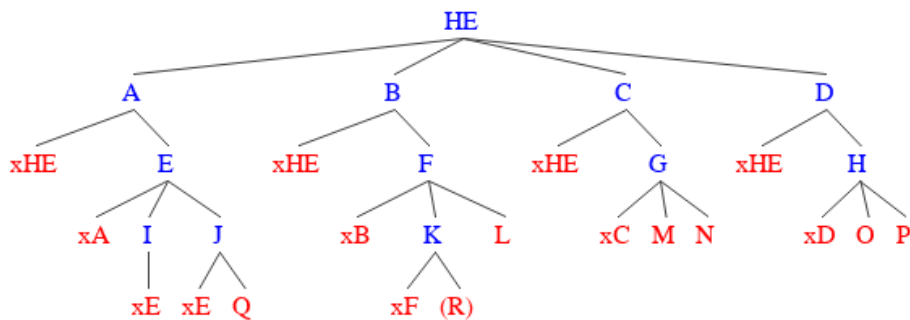
### PROFUNDIDADE RECURSIVA



[http://ironcreek.net/phpsyntaxtree/? \[HE \[A \[xHE\] \[E \[xA\]\[I\[xE\]\]\[J\[xE\]\[Q\]\]\]\] \[B\[xHE\]\[F\[xB\]\[K\[xF\]\[\(R\)\]\]\]\]\]](http://ironcreek.net/phpsyntaxtree/?[HE[A[xHE][E[xA][I[xE]][J[xE][Q]]][B[xHE][F[xB][K[xF][(R)]]]])

- Nós gerados: 18 (se não contar os que já estavam na fronteira=7)
- Máximo de nós em memória: 5 (ao explorar o caminho HAEJQ estão na memória os nós HAEJQ; idem para caminho HBFKR)
- Solução: ir(B)>ir(F)>ir(K)>ir(R)
- Custo = 18

#### LARGURA (CASO 1 – TESTE DE OBJETIVO ANTES DE COLOCAR NA FRONTEIRA)



[http://ironcreek.net/phpsyntaxtree/? \[HE \[A \[xHE\] \[E \[xA\]\[I\[xE\]\]\[J\[xE\]\[Q\]\]\]\] \[B\[xHE\]\[F\[xB\]\[K\[xF\]\[\(R\)\]\]\]\[L\]\] \[C\[xHE\]\[G\[xC\]\[M\]\[N\]\]\] \[D\[xHE\]\[H\[xD\]\[O\]\[P\]\]\]\]\]](http://ironcreek.net/phpsyntaxtree/?[HE[A[xHE][E[xA][I[xE]][J[xE][Q]]][B[xHE][F[xB][K[xF][(R)]]][L]][C[xHE][G[xC][M][N]]][D[xHE][H[xD][O][P]]]])

- Número máximo de nós na memória: 30 (unidirecional: 19; descontam-se os x<Nó>)
- Número de nós gerados: 30 (unidirecional: 19; descontam-se os x<Nó>)
- Solução: ir(B)>ir(F)>ir(K)>ir(R)
- Custo = 18

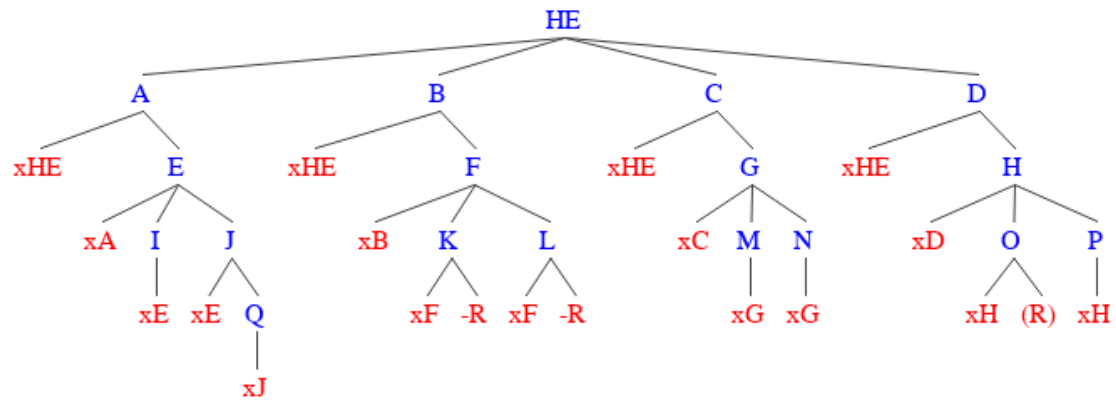
#### BUSCA CUSTO UNIFORME

##### **Legenda**

x<Nó>: nó não incluído na fronteira porque <Nó>.estado já foi explorado

-<Nó>: nó não incluído na fronteira porque <Nó>.estado já está na fronteira e é melhor ou substituído por outro nó melhor

(<Nó>): nó que representa a solução



[HE [A[xHE][E[xA][I[xE]]][J[xE][Q [xJ]]]]][B[xHE][F[xB][K[xF][ -R]]][L[xF][xR]]]]][C[xHE][G[xC][M[xG]]][N[xG]]]]][D[xHE][H[xD][O[xH]][(R)]]][P[xH]]]]]

- **Nós gerados: 38**  
(se descontarmos os nós 17 x<Nó> que representam passagens bidirecionais obtemos 21)
- **Nós em memória: 38** (se descontarmos os nós 17 x<Nó> obtemos 21)
  - Explicação para os nós etiquetados com -<Nó>:
    - HE-B-F-K-R estava na fronteira com custo **18**
    - HE-B-F-L-R foi gerado, mas não incluído na fronteira porque seu custo era **20** (portanto maior do que o R que já estava na fronteira)
    - HE-D-H-O-R foi gerado com custo **15,5** e substituiu o HE-B-F-K-R na fronteira
    - Para fins de padronização da contagem, consideramos que os todos estes nós R foram incluídos na árvore de busca.
- Solução: ir(D)> ir(H)> ir(O)>ir(R)
- Custo=15,5