* 1. Nesta Interrogação, faria sentido usar um índice **B-Tree**, **agrupado e esparso**, **secundário** sobre a tabela **proposta\_de\_correção** em **data\_hora**. Devido ao facto de se tratar de uma seleção com alcance, usamos um índice em árvore. Como é expectável que a interrogação acima devolva mais do que 10% do total de registos da tabela, usamos um índice agrupado para podermos tirar vantagem das suas propriedades e minimizar o número de páginas em memória acedidas (caso contrário seria de eficiência comparável com uma seleção e procura de o ficheiro na integra, que não beneficia da existência de índices).
  2. Nesta interrogação, faria sentido usar um índice **B-Tree, desagrupado e denso, secundário** sobre a tabela, **proposta\_de\_correção** em **data\_hora**. Mais uma vez escolhemos, índice em árvore por se tratar de uma seleção com alcance. No entanto, como nesta alínea é expectável apenas que a interrogação acima devolva menos de 0.001% do total dos registos em tabela, consideramos que não haverá proveito suficiente (embora houvesse de facto proveito na mesma) que justifique adotar um índice agrupado, com os seus altos custos.

1. Nesta interrogação faria sentido usar um índice **Hash**, **desagrupado e denso, primário** (pois inclui a *primary key)*sobre a tabela **incidência** em **anomalia\_id.** Uma vez que se trata de uma seleção de igualdade, o índice mais eficiente será o hash, que beneficia desta situação.

3.1) Nesta Interrogação, faria sentido usar um índice **B-Tree**, **agrupado e esparso**, **, primário** (pois inclui a *primary key)*sobre a tabela **correção** em **anomalia\_id**. Sendo que se trata de uma seleção com alcance, usamos um índice em árvore. Como é expectável que a interrogação acima devolva mais do que 10% do total de registos da tabela, usamos um índice agrupado tira vantagem das suas propriedades e minimiza o número de páginas em memória acedidas (caso contrário seria de eficiência comparável com uma seleção e procura de o ficheiro na integra, que não beneficia da existência de índices).

3.2) Nesta interrogação, faria sentido usar um índice **B-Tree, desagrupado e denso, , primário** (pois inclui a *primary key)*sobre a tabela, **correção** em **anomalia\_id**. Mais uma vez escolhemos, índice em árvore por se tratar de uma seleção com alcance. No entanto, como nesta alínea é expectável apenas que a interrogação acima devolva menos de 0.001% do total dos registos em tabela, consideramos que não haverá proveito suficiente (embora houvesse de facto proveito na mesma) que justifique adotar um índice agrupado, com os seus altos custos.

4) Nesta interrogação, faria sentido usar um índice **B-Tree Composto, agrupado e esparso, secundário** sobre a tabela **anomalia** com a chave composta **<tem\_anomalia\_redação, ts, lingua>**. Como nos casos acima escolhemos um índice em arvore para seleções com alcance, pois estas excelam para estes casos. Será um índice composto para que seja possível tirar proveito dele em todas as comparações da interrogação. Em relação à ordem, optámos em colocar tem\_anomalia\_redação em primeiro pois é uma seleção sem alcance, o que nos pode levar a ignorar 50% dos dados (caso os dados estejam igualmente distribuídos entre os valores de True e False nesta variável) em apenas uma comparação, o que não seria possível caso não estivesse em primeiro lugar do índice. Como esta alínea não fornece nenhuma informação sobre a percentagem de dados expectável pela interrogação, adotamos um índice agrupado, pois este terá a melhor *performance* caso seja necessário retirar os dados de vários elementos. No entanto, face mais informações e consoante a necessidade de outros índices (uma vez que só pode haver um índice agrupado por ficheiro), pode-se também adotar um índice desagrupado, dando prioridade a outras interrogações.