* 1. **Motivação:** Tentar diminuir a quantidade de falta de páginas na memória principal, evitando assim que sejam realizados acessos desnecessários a memória secundária. Utiliza a verificação de 2 bits já presentes.
  2. **Objetivo:** Apresentar o algoritmo de substituição NRU que diferentemente do FIFO utiliza um critério de troca que leva em consideração os bits R e M, cada bit indica respetivamente se o conteúdo daquela página foi acessado e/ou modificado na tabela de páginas.
  3. **Detalhamento** Com isso será executado os seguintes passos:
     1. Carregar o arquivo de arquitetura 7.
     2. Escrever as classes no quadro
        1. **Classe 0:** Não referenciada, Não modificada.
        2. **Classe 1:** Não referenciada, Modificada.
        3. **Classe 2:** Referenciada, Não modificada.
        4. **Classe 3:** Referenciada, Modificada.
     3. Acompanhar passo a passo a execução do processo
     4. Apresentar para o aluno as mudanças nos bits R M.
     5. A cada acesso de leitura ou escrita mostrar no campo SUBs a mudança da classe.
     6. Apresentando as trocas de páginas que ocorrerá nas páginas com menor classe.
     7. Realizar a execução até o final do processo.
  4. **Arquivos**
     1. **Arquitetura**: Architecture-07-MM-16-VM(PS-4-DM-16-RA-NRU)-TLB(none)
     2. **Rastro:** TR\_5\_read\_and\_write\_20\_rand\_PS\_18
  5. **Pontos a destacar** 
     1. O resultado da primeira execução é**:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Memória / Taxas | Acessos de leitura | Acessos de escrita | Page fault  (Page table) | **Tempo total** |
| Page Table | 20 | 2 | 1 | **240** |
| Principal | 36 | 14 |  | **640** |
| Disk | 1 | 1 |  | **300** |
|  |  |  |  | **TT: 940** |