

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**  
**DEPARTAMENTO ACADÊMICO DA COMPUTAÇÃO**  
**ENGENHARIA DE COMPUTAÇÃO**

**DAVI SHINJI MOTA KAWASAKI**  
**HIGOR AUGUSTO BASSI ROZAN**  
**JOÃO VITOR BERTONCINI**  
**LUAN SILVA**  
**VINÍCIUS DRAGO ROMANO**

**IMPLEMENTAÇÃO DO ALGORITMO NESTED LOOP BLOCADO**  
**BUFFERIZADO**

**APS DE BANCO DE DADOS II**

**CORNÉLIO PROCÓPIO**

**2016**

**DAVI SHINJI MOTA KAWASAKI**  
**HIGOR AUGUSTO BASSI ROZAN**  
**JOÃO VITOR BERTONCINI**  
**LUAN ANDRYL SILVA**  
**VINÍCIUS DRAGO ROMANO**

**IMPLEMENTAÇÃO DO ALGORITMO NESTED LOOP BLOCADO  
BUFFERIZADO**

APS desenvolvida para obter nota parcial da disciplina de Banco de Dados II do Curso Engenharia de Computação da Universidade Tecnológica Federal do Paraná ministrada pelo Prof. Pedro Henrique Bugatti.

**CORNÉLIO PROCÓPIO**

**2016**

## 1 ESPECIFICAÇÃO DO PROBLEMA

O problema consiste na operação de junção de duas tabelas através de um algoritmo que simule a operação em SGBDs. O algoritmo utilizado para essa executar a operação desejada foi o de nested loop bloqueado bufferizado, a qual é realizada por loops aninhados através da utilização de buffers para otimização do processamento. O buffer, nesse caso, é uma estrutura de dados que contém uma lista de páginas, e o seu tamanho é dado por entradas pelos parâmetros.

A junção será dada por uma coluna em comum entre duas tabelas, onde cada leitura será dada por blocos de buffer até o final da página, sendo que para cada bloco as leituras dos dados serão realizadas e armazenadas em um vetor final.

As tabelas escolhidas foram uma de departamento e outra de usuários, sendo que a de departamento armazena a chave primária do código e os nomes dos departamentos; já a de usuários armazena a chave primária do código do usuário, seu nome e sobrenome, e por fim a chave estrangeira do código de departamento.

Para o desenvolvimento do algoritmo foi escolhido a linguagem de programação Python 3.5 pela sua facilidade sintática e de manipulação, além da biblioteca python-tabulate, a qual foi utilizada para apresentação, manipulação e impressão de tabelas e listas.

## 2 EXPERIMENTOS

Para os testes respectivos à implementação do algoritmo, foram executados experimentos variando o tamanho do buffer (em Mb) em função do tempo de execução (em ms). A partir do código implementado, variou-se portanto o tamanho do buffer até 1000 Mb, chegando a tempos relevantes de 0,0025 ms, conforme representado na figura abaixo, que mostra a relação de buffer por tempo para o algoritmo desenvolvido pelo grupo:

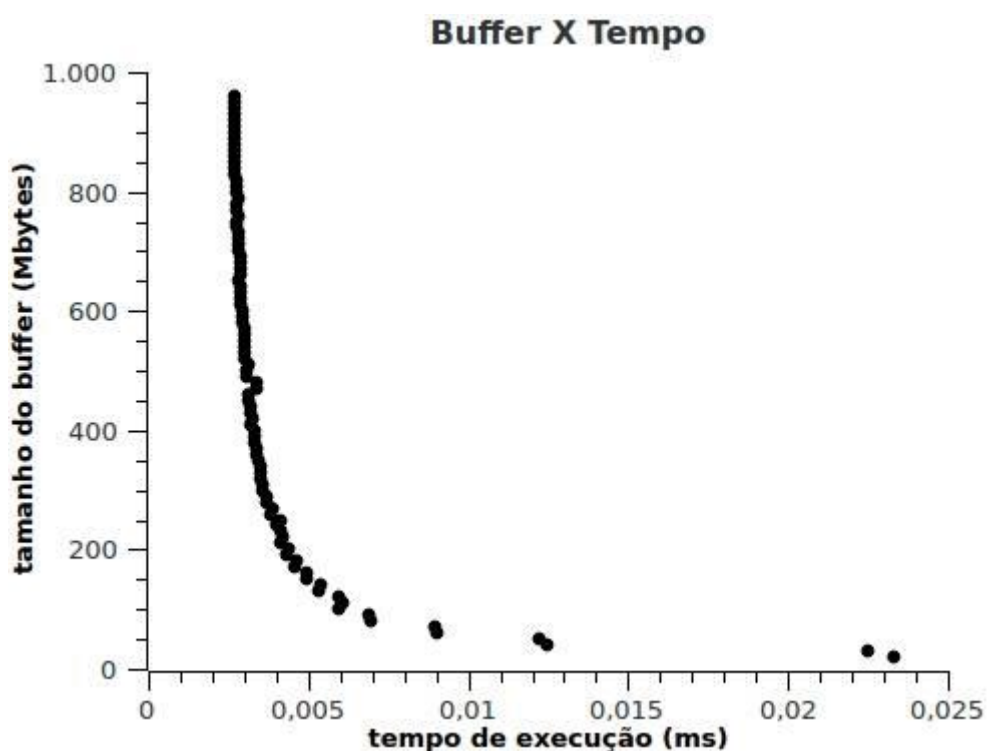


Figura 1 - Experimento realizado aumentando e diminuindo o tamanho dos buffers.  
Fonte: Autoria própria.

Conforme pode-se verificar, o tempo de execução decaiu conforme o tamanho do buffer foi aumentado, dado que o mesmo foi quebrado em vários blocos menores para execução do loop aninhado do algoritmo.