



Objetivo

Apresentar, mesmo que de forma introdutória, o impacto do pipeline de códigos escritos em C com a técnica de desenrolar laços de repetição.

Requisitos

- código fonte em LP C `prodEscOR.c` anexado
- ferramenta profiler `gprof`
-

Roteiro

1. Explorar o código fonte em LP C chamado `prodEscOR.c`
2. Compilar (simples) com `gcc -O0 prodEscOR.c -o prodEscOR.x`
3. Executar 5x com `$ time ./prodEscOR.c`
4. Anotar os tempos `user` e calcular a média
5. Compilar (profiler) com `gcc -O0 -pg prodEscOR.c -o prodEscOR.x`
6. Executar uma única vez para gerar o arquivo profiler `gmon.out` com `$./prodEscOR.x`
7. Executar profiler com `$ gprof -p -b prodEscOR.x gmon.out > a_prodEscOR.txt`
8. Abrir e anotar os tempos `self-seconds` para as funções `inicializarVetor` e `prodEsc`
9. Gerar uma cópia do arquivo `prodEscOR.c` com o nome `prodEscLP01.c`
10. Desenrolar o laço da função `inicializarVetor` para 4 níveis

31.	for (int i = 0; i < N; i+=4) {
32.	v[i] = i;
33.	v[i+1] = i+1;
34.	v[i+2] = i+2;
35.	v[i+3] = i+3;
36.	}

11. Repetir os passos 3. até 8. para `prodEscLP01.x`
12. Gerar uma cópia do arquivo `prodEscLP01.c` com o nome `prodEscLP02.c`
13. **ATIVIDADE:** Desenrolar o laço da função `prodEscalar` para 4 níveis
CUIDADO: não esquecer dos HAZARDS do pipeline!
14. Usando o código `prodEscOR.c`, repita os passos 3. e 4., considerando otimização -O1 e -O2

Entrega

- Documento .pdf contendo a planilha de anotações e o código da função `prodEscalar`
 - Nome do documento: `[OAC P06] nomeSobrenome1 nomeSobrenome2.pdf`
- via [Teams](#) até a data definida na Tarefa