



## Universidade de São Paulo

Instituto de Ciências Matemáticas e de Computação

Departamento de Ciências da Computação

SCC0122 - Estruturas de Dados (2025)

Professora: Mirela Teixeira Cazzolato

Monitores: Lucas Lombardi Castro e Leonardo Rossi Luiz

# Trabalho 04 - Fila de Impressão do ICMC

## 1. Introdução

O gerente do laboratório de computação do ICMC está enfrentando um problema: nos horários de pico, dezenas de alunos enviam trabalhos para a única impressora do andar, gerando uma confusão de documentos. Para organizar o caos e garantir que os trabalhos sejam impressos na ordem correta de chegada (FIFO - "First-In, First-Out"), ele precisa de um sistema de gerenciamento de impressão. Para validar a ideia, sua equipe foi contratada para desenvolver um protótipo funcional deste sistema por linha de comando!

## 2. Descrição do Problema

O programa deverá ler uma sequência de comandos, um por linha. Os comandos podem ser para adicionar um novo documento à fila de impressão, imprimir o próximo documento da fila, listar os documentos pendentes ou encerrar o programa. Após o comando de impressão, o programa deve exibir qual documento foi "impresso". O programa deve continuar executando até que o usuário insira o comando off. Ao receber o comando off, o simulador deve listar os trabalhos restantes na fila e então encerrar a aplicação.

Objetivo	Comando	Descrição da Operação
<b>Adicionar Documento</b>	<code>add &lt;nome&gt; &lt;páginas&gt;&lt;cor&gt; &lt;formato_pagina&gt;</code>	Lê o nome de um arquivo (string), número de páginas (int), cor (string) e formato da página (string) e o insere no final da fila. Nenhuma saída é impressa nesta etapa.
<b>Imprimir Próximo</b>	<code>print</code>	Remove o documento do início da fila e imprime uma mensagem confirmando a impressão. Se a fila estiver vazia, deve informar o usuário.
<b>Listar Fila</b>	<code>list</code>	Exibe todos os documentos na fila, do início para o fim, sem removê-los.
<b>Encerrar</b>	<code>off</code>	Imprime todo o conteúdo restante da fila, do início para o fim, e encerra a execução do programa.

### 3. Especificações

Atenção! Para a avaliação do projeto, não será considerada apenas a nota do Run Codes, mas também o cumprimento das seguintes especificações:

- O código deve estar na linguagem C.
- O código deve ser organizado usando um **TAD para a Fila** e um **TAD para arquivo** (contendo *nome\_arquivo*, *numero\_paginas*, *cor*, *formato\_pagina*), seguindo a estrutura vista em aula.
- Uma sugestão é dividir o projeto em três arquivos: (**main.c**, **fila.c** e **fila.h**).
- A legibilidade do código é um fator importante. Use nomes de variáveis e funções que sejam claras e comente o código de forma apropriada.

### 4. Exemplos de Entrada e Saída

Exemplo de Entrada	Exemplo de Saída
add relatorio.pdf 15 pb .pdf add codigo.c 3 pb .c print list add tese.docx 120 cor .docx off	Imprimindo: relatorio.pdf (15 paginas, pb, .pdf) Fila: [codigo.c (3 paginas, pb, .c)]  Fila final: [codigo.c (3 paginas, pb, .c), tese.docx (120 paginas, cor, .docx)]

Exemplo de Entrada	Exemplo de Saída
print add trabalho_ed1.zip 1 pb .zip add prova_calc.pdf 4 cor .pdf print print off	Fila de impressao vazia.  Imprimindo: trabalho_ed1.zip (1 pagina, pb, .zip) Imprimindo: prova_calc.pdf (4 paginas, cor, .pdf) Fila final: []

### 5. Submissão

1. **Envie** seu código fonte para o run.codes. Para isso, compacte (.zip), os arquivos, junto com um documento **Makefile**. (Será executado o comando “make all” para compilar seu código, e o comando “make run” para executar o seu código)
2. **Tire dúvidas!** Se tiver dúvida sobre algo, mande um email para os monitores, ou participe da monitoria presencial.