

# LAB 2

# Context

INF01142 - Sistemas Operacionais IN

Prof.: Alexandre Carissimi

Monitor: Eduardo Roloff (mestrando)

# Roteiro

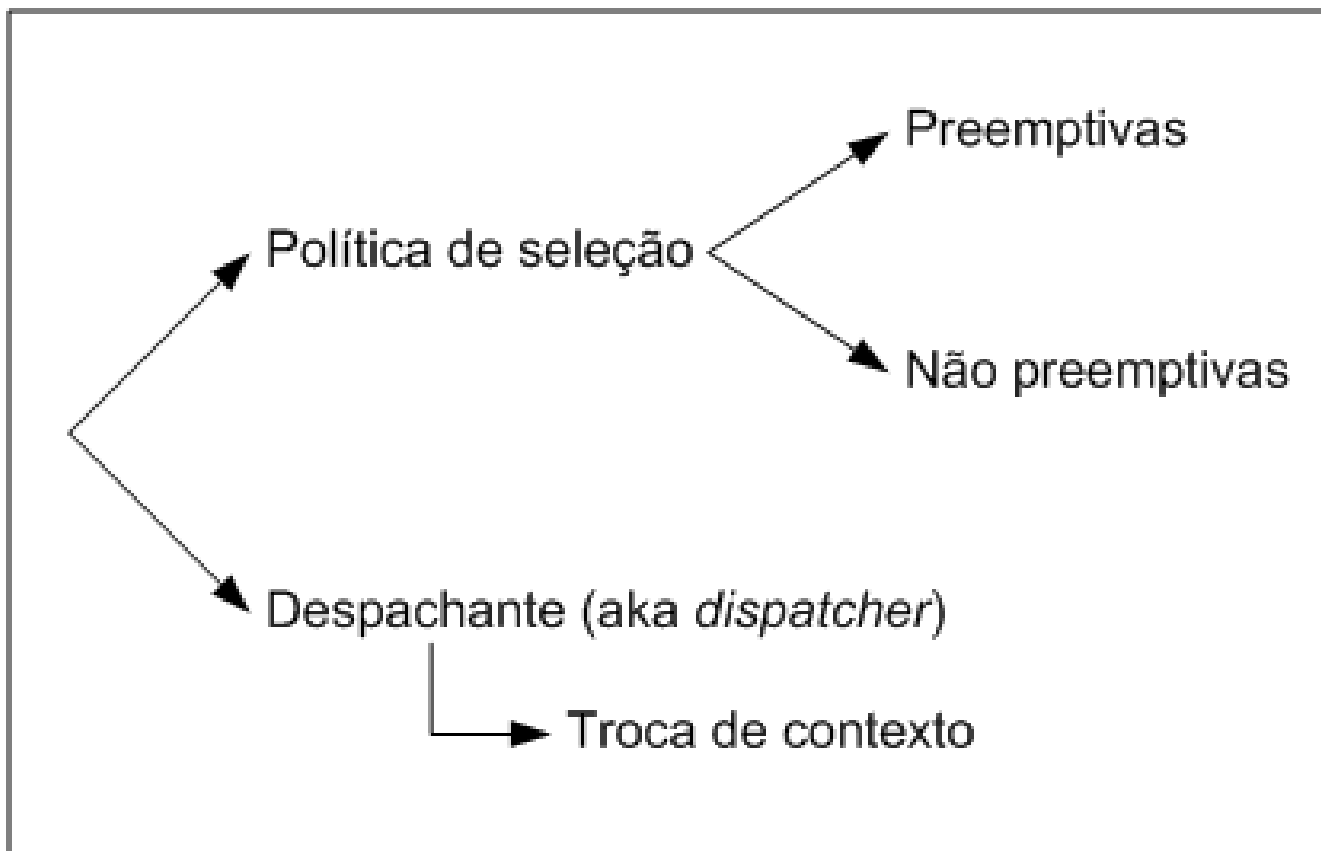
- Ferramental
- Revisão
- Exercícios

# Ferramental

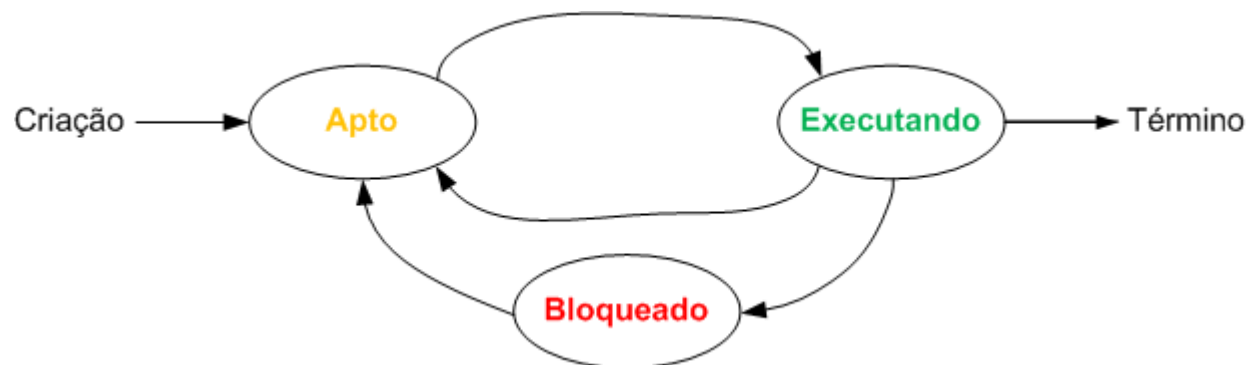
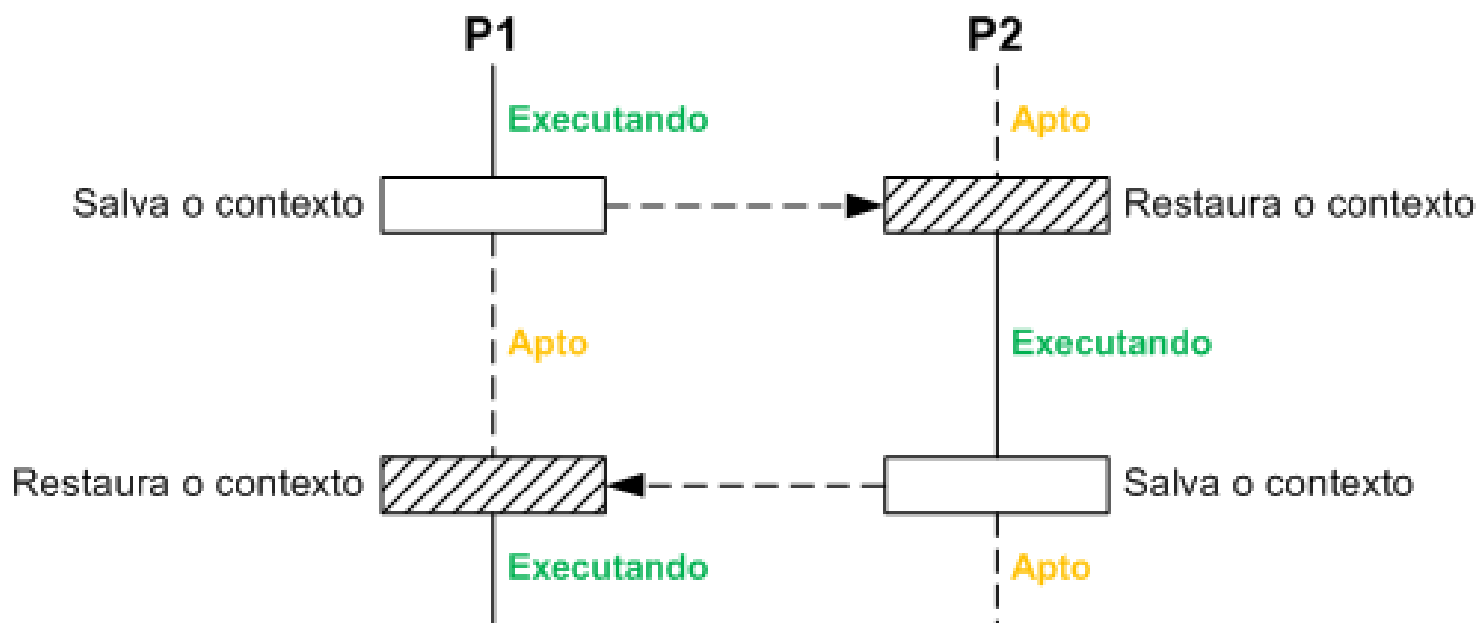
- Conhecer um bom editor de texto
- Saber usar o “man”
- Conhecer uma linguagem de programação razoável
  - Isso significa C
- Yahoo!, Google e Bing são alguns de seus amigos

# Revisão

- Mecanismo de escalonamento



# Revisão



# Revisão

- Contexto
  - Contexto de execução de múltiplos fluxos
  - Permite ao usuário controlar o contexto de execução

```
#include <ucontext.h>
```

```
typedef struct ucontext {  
    struct ucontext *uc_link;  
    sigset_t uc_sigmask;  
    stack_t uc_stack;  
    mcontext_t uc_mcontext;  
    ...  
} ucontext_t;
```

# Makecontext

```
void makecontext(ucontext_t *ucp,  
                void (*func)(), int argc, ...);
```

- Modifica o contexto apontado por **ucp**
- É necessário inicializar a estrutura **antes**
  - Pilha (**uc\_stack**)
  - Próximo contexto a ser executado (**uc\_link**)
- **Retorno**
  - Função **void**
- Mais detalhes em: `~$man makecontext`

# Swapcontext

```
int swapcontext(ucontext_t *oucp,  
                ucontext_t *ucp) ;
```

- Faz a troca de contexto
- Salva o contexto atual em **oucp**
- Ativa o contexto apontado por **ucp**
- **Retorno**
  - -1 em caso de erro
  - Não há retorno em caso de sucesso
- Mais detalhes em: `~$man swapcontext`



# Getcontext

```
int getcontext(ucontext_t *ucp) ;
```

- Inicializa **ucp** com o contexto atual
  - Contexto atual = o que está em execução

- **Retorno**

- -1 em caso de erro
  - 0 em caso de sucesso
- Mais detalhes em: `~$man getcontext`

# Setcontext

```
int setcontext(const ucontext_t *ucp) ;
```

- Restaura o contexto apontado por **ucp**
  - Precisa ter sido inicializado antes

- **Retorno**

- -1 em caso de erro
  - Não retorna em caso de sucesso
- Mais detalhes em: ~\$man getcontext

# Exercícios

- Exercício 1
  - Análise de código fonte
    - Analisar o fluxo de execução do código lab02.c
- Exercício 2
  - Programação
    - Programa com duas funções que use troca de contexto
    - Uma função imprime pares e outra ímpares
    - Função main controla o fluxo e testa final da execução
    - **Proibido o uso de globais (exceto os contextos)**
    - **Proibida a chamada direta de funções pela main**