Pergunta 1:

1. PrintOrd:

a. <u>waitPoint</u> – ponto de sincronização em que o pedido de impressão espera pelo ID da impressora para a qual irá enviar o pedido (Não precisa ser inicializado no inicio)

2. waitQueue:

- a. <u>access</u> serve para garantir a exclusão mútua no acesso à fila de espera (valor inicial = 1).
- b. <u>notEmpty</u> indica aos processos consumidores se a fila está vazia ou não (valor inicial = 0).
 - i. Consumidor faz down
 - ii. Produtor faz up
- c. <u>notFull</u> indica aos processos produtores se a fila está cheia ou não (valor inicial = NORDS).
 - i. Produtor faz down
 - ii. Consumidor faz up

3. PrinterPool:

a. <u>access</u> – serve para garantir a exclusão mútua no acesso ao parque de impressoras (valor inicial = 1).

Pergunta 2:

EXTRA (queue_in):

Pergunta 3:

```
void printing_server (int id)
{
    PrintOrd* ordem;

    while ((queue -> notEmpty) != 0)
    {
        ordem = queue_out(&queue);
        (ordem -> id) = id;
        sendo_to_printer(ordem -> filename);
    }
}
```

Pergunta 4:

Esta acho que é um bocado à vontade do freguês, desde que bem explicada.

```
#define UMA_FACE 1
#define DUPLA_FACE 2
```

Acrescentar o campo print na estrutura PrintOrd e Printer. Que pode tomar os valores UMA_FACE ou DUPLA_FACE.

Acrescentar duas variáveis na estrutura PrinterPool que indicam quantas impressoras frente (FRENTE) e frente e verso (FRENTE_VERSO) há no parque de impressoras.