

Trabalho prático 2

Algoritmo e Estrutura de dados I – 22.2

Mudanças necessárias

Objetivo do trabalho

O presente trabalho tem por objetivo explorar estruturas de dados exploradas em sala de aula no início deste semestre. Para tanto, deve-se desenvolver um software que implementa uma solução para o problema detalhado a seguir.

Detalhamento

O dono de um mercado está assustado com o sumiço dos seus clientes e entende que novos tempos chegaram. Novos tempos requerem novas soluções. A característica dos clientes do mercado agora é o de fazer pedidos pelo *whatsapp*, mas o dono do empreendimento não está conseguindo se organizar. Ele está se perdendo entre os pedidos, a cobrança e as entregas. Então ele recorreu a você para ajudar nesta tarefa.

O mercado possui um único canal de recepção de mensagens com os pedidos dos clientes. Estes pedidos devem ter sua ordem de atendimento respeitada pela ordem em que foram enviadas. Cada cliente envia um pedido com uma lista de produtos que deseja. Há 3 “separadores”, pessoas responsáveis por pegar o pedido e coletar os produtos no mercado. Como cada produto encontra-se em locais diferentes do mercado, dependendo do tamanho da lista, a coleta de material leva mais ou menos tempo. Cabe salientar que a lista do pedido é um recurso fictício. Sendo assim, pode-se listar uma quantidade de produtos, ao invés de listar os produtos especificamente. A coleta de cada produto pode levar um tempo aleatório num intervalo estipulado pelos autores. Ao finalizarem a coleta dos produtos, cada “separador” coloca o pedido em uma fila de pedidos pronto para a entrega.

A entrega é feita por três entregadores, que levam um tempo aleatório para fazer a entrega de cada pedido. Inconformados com eventuais demoras, alguns clientes desistem de sua compra, a menos que ele já esteja a caminho, ou seja, esteja com o pessoal da entrega.

Construa um sistema que considere rodadas para mapear toda esta trajetória apontada. Em uma rodada (a) um pedido pode entrar no sistema, (b) separadores podem atender pedidos e/ou coletar os produtos do pedido e/ou inserir na fila de pedidos prontos e (c) entregadores livres podem coletar pedidos para serem entregues. A cada rodada, cria a probabilidade de um pedido entrar pelo aplicativo de mensagem.

Ao entrar, caso algum separador esteja disponível, este já pode iniciar a coleta dos produtos, que deve iniciar no ciclo. Cada produto da lista leva um ciclo para ser coletado. Após todos serem coletados, este deve ser adicionado na fila de pedidos prontos para entrega. Caso um entregador esteja disponível, este já pode iniciar a entrega, que pode levar de 4 a 8 rodadas para fazer a entrega e estar disponível para coletar um novo pedido.

O dono do estabelecimento precisa saber o desempenho deste esquema que montou. Então é importante saber quantos pedidos deram entrada, quantos foram bem sucedidos, quantos quase foram cancelados e quantos efetivamente foram cancelados. Como seus colaboradores ganham por comissão, precisa saber quais separaram mais pedidos e quais fizeram mais entregas. Precisa saber ainda o tempo médio de cada pedido na fila e o tempo que levou mais tempo para ser atendido.

O programa deve fornecer uma interface gráfica com o usuário que permita rodar por um determinado número de ciclos. A interface pode ser elaborada diretamente na console ou a partir do uso de algum recurso mais elaborado.

Considerações finais

O trabalho pode ser desenvolvido em duplas formada por alunos de uma mesma turma da disciplina. A tecnologia a ser empregada na elaboração do trabalho é de escolha da dupla, mas deve-se limitar as linguagens Java, C/C++ ou Python.

Além da solução em software elaborada, deve-se enviar um relatório contendo detalhes da implementação bem como um passo a passo para a compilação/execução e exemplos de execução da solução utilizadas para validar o funcionamento.

O trabalho deve ser encaminhado na sala de entrega do moodle, contendo os códigos fonte e um relatório breve descrevendo as classes que foram idealizadas e a forma de funcionamento do programa elaborado. Todos os arquivos devem ser compactados em um arquivo zip, que deve ter o nome e sobrenome do autor (e.g. JosePenteado_MariaErnestina.zip).

Por fim, cabe reforçar que a percepção de plágio de trabalhos acarretará grau zero para todos os envolvidos. Portanto, não compartilhe o seu código. Caso perceba que seu trabalho foi plagiado, notifique imediatamente o professor.