PRO5802 - Programação de Produção Intermitente (2023)

Prof. Daniel de Oliveira Mota

Data	Aula	Conteúdo
19/09	1a 1b	Apresentação do curso Revisão Python
26/09	2a 2b	Teoria (revisão) de Grafos Aspectos avançados (funções e conjuntos) de Python
03/10	3a 3b	Caracterização do problema de scheduling
10/10	4a 4b	Implementação de heuristicas no caixeiro viajante
17/10	5a 5b	Casos: Flowshop Construção do ScheduleTable da solução
24/10	6a 6b	Apresentação conjunto 1 de artigos (Casos: Flowshop) / Discussão de projeto / Definição de escopo
31/10	7a 7b	Casos: Flowshop Implementação de heurísticas clássicas (Moore e Johnson) / NEH
07/11	8a 8b	Casos: Jobshop Implementação de MetaHeurísticas (Simulated Annealing)
14/11	9a 9b	Casos: Discussão artigo “Literature Review” Oficina de desenvolvimento (Google OR-Tools)
21/11	11a 11b	Atendimento dos grupos em dúvidas finais
28/11	12a 12b	Apresentação projetos e fechamento da disciplina

Apresentação do assunto

- Exemplo 1.1.1
- Exemplo 1.1.2
- Exemplo 1.1.3
- Exemplo 1.1.4
- *Scheduling em manufatura
- *Scheduling em serviços

* extra

Example 1.1.1 (A Paper Bag Factory). Consider a factory that produces paper bags for cement, charcoal, dog food, and so on. The basic raw material for such an operation are rolls of paper. The production process consists of three stages: the printing of the logo, the gluing of the side of the bag, and the sewing of one end or both ends of the bag. Each stage consists of a number of machines which are not necessarily identical. The machines at a stage may differ slightly in the speed at which they operate, the number of colors they can print, or the size of bag they can produce. Each production order indicates a given quantity of a specific bag that has to be produced and shipped by a committed shipping date or due date. The processing times for the different operations are proportional to the size of the order, i.e., the number of bags ordered.

A late delivery implies a penalty in the form of loss of goodwill and the magnitude of the penalty depends on the importance of the order or the client and the tardiness of the delivery. One of the objectives of the scheduling system is to minimize the sum of these penalties.

When a machine is switched over from one type of bag to another a setup is required. The length of the setup time on the machine depends on the similarities between the two consecutive orders (the number of colors in common, the differences in bag size, and so on). An important objective of the scheduling system is the minimization of the total time spent on setups. ||

Example 1.1.2 (A Semiconductor Manufacturing Facility). Semiconductors are manufactured in highly specialized facilities. This is the case with memory chips as well as with microprocessors. The production process in these facilities usually consists of four phases: wafer fabrication, wafer probe, assembly or packaging, and final testing.

Wafer fabrication is technologically the most complex phase. Layers of metal and wafer material are built up in patterns on wafers of silicon or gallium arsenide to produce the circuitry. Each layer requires a number of operations, which typically include: (i) cleaning, (ii) oxidation, deposition, and metallization, (iii) lithography, (iv) etching, (v) ion implantation, (vi) photoresist stripping, and (vii) inspection and measurement. Because it consists of various layers, each wafer has to undergo these operations several times. Thus, there is a significant amount of recirculation in the process. Wafers move through the facility in lots of 24. Some machines may require setups to prepare them for incoming jobs; the setup time often depends on the configurations of the lot just completed and the lot about to start.

The number of orders in the production process is often in the hundreds and each has its own release date and a committed shipping or due date. The scheduler's objective is to meet as many of the committed shipping dates as possible, while maximizing throughput. The latter goal is achieved by maximizing equipment utilization, especially of the bottleneck machines, requiring thus a minimization of idle times and setup times. ||

Example 1.1.3 (Gate Assignments at an Airport). Consider an airline terminal at a major airport. There are dozens of gates and hundreds of planes arriving and departing each day. The gates are not all identical and neither are the planes. Some of the gates are in locations with a lot of space where large planes (widebodies) can be accommodated easily. Other gates are in locations where it is difficult to bring in the planes; certain planes may actually have to be towed to their gates.

Planes arrive and depart according to a certain schedule. However, the schedule is subject to a certain amount of randomness, which may be weather related or caused by unforeseen events at other airports. During the time that a plane occupies a gate the arriving passengers have to be deplaned, the plane has to be serviced and the departing passengers have to be boarded. The scheduled departure time can be viewed as a due date and the airline's performance is measured accordingly. However, if it is known in advance that the plane cannot land at the next airport because of anticipated congestion at its scheduled arrival time, then the plane does not take off (such a policy is followed to conserve fuel). If a plane is not allowed to take off, operating policies usually prescribe that passengers remain in the terminal rather than on the plane. If boarding is postponed, a plane may remain at a gate for an extended period of time, thus preventing other planes from using that gate.

The scheduler has to assign planes to gates in such a way that the assignment is physically feasible while optimizing a number of objectives. This implies that the scheduler has to assign planes to suitable gates that are available at the respective arrival times. The objectives include minimization of work for airline personnel and minimization of airplane delays.

In this scenario the gates are the resources and the handling and servicing of the planes are the tasks. The arrival of a plane at a gate represents the starting time of a task and the departure represents its completion time. ||

Example 1.1.4 (Scheduling Tasks in a Central Processing Unit (CPU)). One of the functions of a multi-tasking computer operating system is to schedule the time that the CPU devotes to the different programs that have to be executed. The exact processing times are usually not known in advance. However, the distribution of these random processing times may be known in advance, including their means and their variances. In addition, each task usually has a certain priority level (the operating system typically allows operators and users to specify the priority level or weight of each task). In such case, the objective is to minimize the expected sum of the weighted completion times of all tasks.

To avoid the situation where relatively short tasks remain in the system for a long time waiting for much longer tasks that have a higher priority, the operating system "slices" each task into little pieces. The operating system then rotates these slices on the CPU so that in any given time interval, the CPU spends some amount of time on each task. This way, if by chance the processing time of one of the tasks is very short, the task will be able to leave the system relatively quickly.

An interruption of the processing of a task is often referred to as a *preemption.* It is clear that the optimal policy in such an environment makes heavy use of preemptions.

||