

Escalonador Inteligente

Carla Nobrega, Clarissa Lippi, Eduardo Machado, Hans Hudson, Robson Luiz, Enzo Tiezzi, Igor Moreira, Newton Delbuque, Marco Aurélio, Flávio Augusto, Thiago Souza, Caio Zanelato, Professor Fabricio Barth

Faculdade de Tecnologia Bandeirantes – BandTec

Rua Estela, 268 – 04011-001 – São Paulo – SP – Brasil

1. Contexto

A faculdade BandTec oferece em sua grade curricular um apoio psicopedagógico com o objetivo de suprir as necessidades de seus alunos além do âmbito técnico. O Programa H, como é chamado, auxilia os alunos a desenvolver qualidades humanas como liderança, trabalho em grupo, autoconhecimento, entre outros quesitos que, assim como o conhecimento técnico, são indispensáveis a um profissional de destaque.

Uma das formas de orientar e guiar os alunos em seu desenvolvimento é através de alguns treinamentos organizados pela equipe do programa H, estes treinamentos tem o objetivo de auxiliar os alunos a melhorar alguma competência não tão desenvolvida.

Dado o grande número de alunos e a quantidade de treinamento que o programa H pode executar por semana, a organização da agenda de treinamento dispende muito tempo para ser elaborada e sua adaptação a mudanças repentinas de disponibilidade acaba tornando-se lenta e trabalhosa.

Para auxiliar os integrantes do Programa H a realizar o seu trabalho diário está sendo desenvolvida, por alunos do 5º semestre de Análise e Desenvolvimento de Sistemas da faculdade BandTec, uma plataforma online com o intuito de automatizar e facilitar a criação e manutenção de indicadores de resultados gerados pelo Programa H.

1. Objetivo

O objetivo do algoritmo é conseguir agendar até quatro treinamentos por semana baseando-se na disponibilidade de horário dos alunos, conseguindo agrupar o maior número de alunos possíveis por treinamento.

1. Solução Proposta

Para a escolha do método de desenvolvimento desse sistema foram feitas comparações entre os dois algoritmos de busca de subida da montanha, que serão descritos a seguir:

**Busca de subida da montanha 1:** É uma técnica que usa um laço repetitivo que se move de forma contínua no sentido do valor crescente da função, ou seja, encosta acima, até alcançar um pico (máximo local em que nenhum vizinho tem valor mais alto). Não mantém uma árvore de busca, e assim a estrutura de dados do nó atual só precisa registrar o estado e o valor de sua função objetivo. A técnica não examina antecipadamente valores de estados além dos vizinhos imediatos do estado corrente.

A subida de encosta frequentemente fica paralisada, pelas seguintes razões (limitações):

* **Máximos locais**: quando o algoritmo alcança a vizinhança de um máximo local, será deslocado para cima em direção ao pico, e ficará paralisado, sem ter para onde ir.
* **Picos**: Os picos resultam em uma sequência de máximos locais que torna muito difícil a navegação para algoritmos gulosos, o que paralisa a busca local.
* **Platôs**: ou planícies. Pode existir um máximo local plano, a partir do qual não existe nenhuma saída para a encosta acima. A busca talvez seja incapaz de encontrar a saída do platô e ficar paralisada.

**Busca de subida da montanha 2:** Essa técnica tenta combinar a subida da montanha 1 (incompleto) com um percurso aleatório (completo, porém ineficiente), para ser eficiente e completo escapando do problema da paralisação da subida da montanha 1. A busca de subida da montanha 2 pode fazer modificações que pioram o estado no momento, para possivelmente melhorá-lo no futuro. [1]

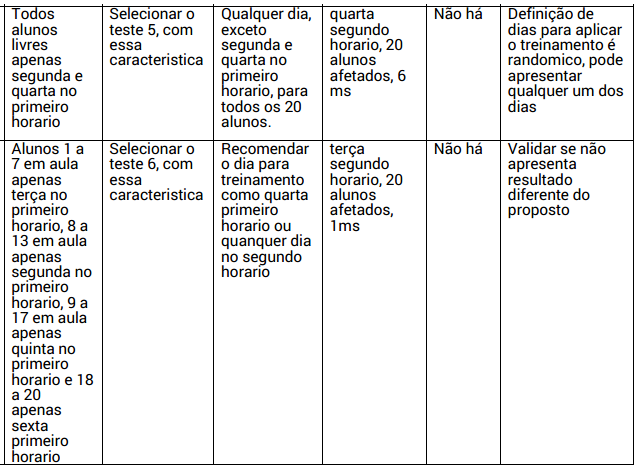
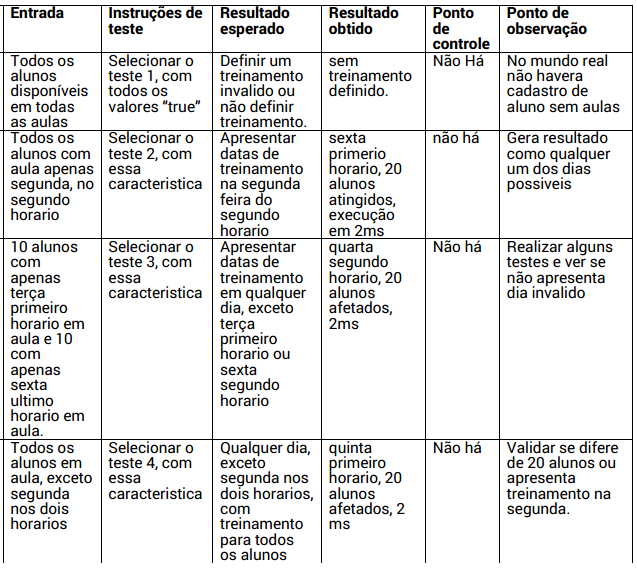
Com base nas comparações o algoritmo escolhido para o desenvolvimento do sistema será o busca de subida da montanha 2, pois se mostrou mais eficiente em encontrar uma solução ótima, que nesse caso seria maximizar a quantidade de alunos por treinamento.

O escalonador usa como parâmetros de entrada uma lista de alunos, e a agenda do aluno e retornará a agenda do treinamento com a quantidade de alunos agendados para aquele determinado treinamento.

1. Validação e Resultados

Para verificar se a aplicação está realmente atendendo a demanda dos objetivos, foram realizados testes no sistema. Realizar testes não consiste simplesmente na geração e execução de casos de teste, mas envolvem também questões de planejamento, gerenciamento e análise de resultados.

A seguir, alguns dos cenários de casos de testes utilizados para validação da aplicação de acordo com o objetivo.



Para esse sistema foram realizados testes unitários que são testes de unidade de um código que tem como objetivo detectar erros ou defeitos em certo nível de classe do sistema. Se trata de um teste funcional.

O teste funcional realizado atendeu os seguintes requisitos:

* Verificar se o sistema respeita aos critérios de apenas 4 treinamentos por semana;
* Verificar se treinamento é agendado apenas quando a quantidade máxima de alunos é escalonada. (Resultado ótimo);
* Verificar se não tiver a quantidade mínima de alunos para o treinamento o treinamento não deverá ser agendado;
* Verificar se o sistema respeita os treinamentos disponíveis X horários livres de alunos para treinamento, juntamente com a quantidade mínima e máxima de alunos por treinamento;

Para garantir que a solução não tenha nenhum problema referente a máximos locais, todos os casos de teste foram executados 20 vezes e sempre retornaram o resultado ótimo.

Foi realizada uma sequência de 20 cenários de testes onde cada teste foi executado 20 veze, totalizando 400 execuções. Onde o tempo médio para resposta foi de 1,44 milissegundos e o desvio padrão foi de 1,33milissegundos.

Alinhado com os objetivos deste projeto, os resultados esperados para o sucesso do mesmo é que o sistema combine (realize um match) entre os horários e datas disponíveis dos alunos que precisarão realizar um curso de capacitação, e seu instrutor. O treinamento é agendado apenas se houver um dia na semana na qual a maior parte dos alunos cadastrados tenham disponibilidade. No entanto, a faculdade pode realizar quatro treinamentos por semana. A definição dos treinamentos é feita nos quatro dias que mais possui alunos disponíveis.

1. Considerações Finais

O objetivo deste trabalho foi propor um sistema de escalonamento levando-se em consideração restrições de agenda e da equipe de treinamento. Para alcançar este objetivo foi utilizado um algoritmo do tipo subida na montanha. Os resultados encontrados comprovam que esta solução é factível e pode ser implantada em ambiente de produção. Todos os cenários de teste validados retornaram configurações ótimas e o tempo de processamento é em media 1,5 milissegundos, o que comprova que o algoritmo pode ser usado para a solução proposta.

1. Referências bibliográficas

[1] NORVIG, PETER,RUSSELL, STUART. **Inteligência Artificial**. 3° ed. Brasil: Elsevier Brasil, 2014.