

Sistema de Venda de Ingressos

Enzo C. da S. Barbosa

UEFS – Universidade Estadual de Feira de Santana

Av. Transnordestina, s/n, Novo Horizonte

Feira de Santana – BA, Brasil – 44036-900

caua7uefs@gmail.com

Abstract. *The Computer Engineering Academic Directory (DA) will hold a show to raise funds, ticket sales will be carried out through a sales system developed in Python. The system collects information about who is selling and who is buying and computes this data to be displayed, quantifying the amount collected, the average age and total for each type of sale and ticket. This program is based on a repeat loop for each sale, where counters and accumulators are used to record responses. Finally, the results are calculated and presented to the user. The program developed solves the problem of this event and can be customized to suit other events.*

Resumo. *O Diretório Acadêmico (DA) de Engenharia de Computação irá realizar um show para arrecadação de recursos para a venda de ingressos será realizada por meio de um sistema de venda desenvolvido em Python. O sistema coleta informações sobre quem está vendendo e quem está comprando e computa esses dados para serem exibidos, quantificando o valor arrecadado, a média de idades e total de cada tipo de venda e ingresso. Este programa é baseado em um laço de repetição para cada venda, onde são usados contadores e acumuladores para registrar as respostas. Por fim, os resultados são calculados e apresentados ao usuário. O programa desenvolvido soluciona o problema deste evento e pode ser customizado para atender outros eventos.*

1. Introdução

O Diretório Acadêmico (DA) de Engenharia de Computação irá realizar um show para arrecadação de recursos, onde serão vendidos ingressos para toda a comunidade. Para ter controle sobre as vendas e poder prestar contas sobre a arrecadação, o DA solicitou aos alunos do 1º semestre a criação de um sistema de venda de ingressos, que atendesse alguns requisitos e que pudesse ser reutilizado em outros eventos.

O objetivo deste relatório é descrever o desenvolvimento de um sistema na linguagem de programação Python, como solução para o problema descrito acima. O sistema serve para realizar a venda de uma determinada quantidade de ingressos, armazenando as informações de cada venda e após a emissão de todos estes ingressos ou o fim do período de vendas, computa todas as informações coletadas e exibe os valores para o usuário como a média de idades, os valores arrecadados e a quantidade de cada tipo de venda realizada.

Para solucionar este problema, foi necessário solucionar algumas questões: 1) Como as questões serão organizadas dentro do sistema; 2) Como permitir que o programa seja customizável; 3) Como implementar no programa o sistema de datas (período das vendas); 4) Como fazer o sistema repetir as vendas; 5) Como encerrar a repetição das vendas; 6) Como registrar as informações de cada venda; 7) Como emitir a cortesia do vendedor comissionado de acordo com os ingressos vendidos por ele; 8) Como realizar tratamentos de erros nas entradas;

A solução está organizada da seguinte forma:

1. As questões são organizadas por prioridade;
2. O usuário poderá inserir os dados customizáveis antes de iniciar as vendas;
3. O sistema de datas foi substituído pela opção de escolha do usuário para encerrar;
4. Foi utilizado uma laço de repetição While;
5. No loop, as vendas encerram caso o critério de ingressos vendidos seja atendido ou caso o usuário escolha encerrar;
6. No loop, ao final de cada venda são incrementadas variáveis contadoras e acumuladoras, que foram declaradas no início do programa;
7. No loop, foi adicionado a opção da venda comissionada para a emissão de cortesia;
8. Em todas as entradas foram utilizados laços de repetição que só permitem entradas válidas.

2. Metodologia

Nesta seção, será descrito os requisitos e funcionalidades do sistema, de acordo com o que foi solicitado no problema, as questões e decisões abordadas durante as sessões tutoriais, a descrição do algoritmo desenvolvido, a ordem de codificação e os sistemas utilizados na elaboração do projeto.

2.1. Requisitos e Funcionalidades

1) Controle: O sistema deve ser capaz de controlar cada venda, indicando a quantidade de ingressos vendidos e quantos ainda restam; 2) Verificação: O sistema deve ser capaz de verificar o tipo de ingresso que está sendo vendido e a disponibilidade; 3) Registro: O sistema deve ser capaz de registrar e armazenar as informações dos tipos de ingresso e venda, e das idades dos compradores; 4) Finalização: O sistema deve ser capaz de finalizar ao ser atendida uma condição ou quando for escolhido a opção de encerrar; 5) Exibição de Resultados: O sistema deve ser capaz de computar todas as informações registradas e exibir para o usuário todas as informações solicitadas.

2.2. Questões e Soluções

1) Como as questões serão organizadas dentro do sistema: As questões foram organizadas por ordem de prioridade e dependência, da seguinte forma: escolha de vendedor (Emissão de Cortesia, Vendedor Comum e Vendedor Comissionado); escolha do tipo de ingresso (Inteira, Meia e Preço Especial); para a escolha de vendedor comissionado existe a opção de tipo de venda (Individual ou Comissionada); para a escolha do tipo de ingresso meia existe a opção de tipo de meia (Estudante ou Outros

Critérios); ao final de todas essas escolhas ocorre o registro da idade, justamente para evitar que a idade seja registrada em caso de ingresso inválido; além disso em todos os menus de escolha existe a opção de cancelar a venda.

2) Como permitir que o programa seja customizável: Antes de iniciar o sistema de repetição das vendas, o usuário deverá inserir o dados do evento desejado, este programa permite a customização da quantidade de ingressos disponíveis, o valor do ingresso, se existe ou não preço especial e em caso de existir, permite customizar o valor do preço especial.

3) Como implementar no programa o sistema de datas (período das vendas): Inicialmente foi dialogado durante as sessões tutoriais 1 e 2, a ideia de adicionar ao sistema o encerramento após o período de venda, porém devido às limitações impostas pelo problema, que não permitia a utilização de bibliotecas do Python que permitem a manipulação de valores de data, foi decidido pela substituição do sistema de datas pela opção do usuário informar quando deseja encerrar o programa, que ocorre durante a primeira parte do laço de repetição, onde o usuário escolhe o tipo de vendedor e existe a opção de encerrar as vendas.

4) Como fazer o sistema repetir as vendas: Foi utilizado um laço de repetição While principal, que repete a venda dos ingressos até que o critério de validação seja atendido ou que seja selecionada a opção de encerrar.

5) Como encerrar a repetição das vendas: No laço de repetição principal a condição de fim do loop é que a quantidade de ingressos seja igual a zero, valor esse que é decrementado a cada repetição do loop. O loop também pode ser encerrado por meio da escolha da opção de encerrar, que leva o loop a um comando break e o encerra.

6) Como registrar as informações de cada venda: Ao final de cada repetição do loop principal é incrementado ao valor da variável acumuladora ou contadora correspondente ao tipo de venda que está sendo realizada.

7) Como emitir a cortesia do vendedor comissionado de acordo com os ingressos vendidos por ele: O problema especifica que para receber uma cortesia, o vendedor comissionado deverá trazer todos o ingressos necessário de uma única vez, sendo assim foi implementado a opção de realizar a venda comissionada, onde de uma única vez é realizado a venda quantidade de ingressos necessários para obter a cortesia.

8) Como realizar tratamentos de erros nas entradas: Todas as entradas deste programa foram feitas para receberem valores numéricos, caso o usuário digite um valor não numérico como letras, símbolos ou espaços vazios, o programa entrará em um laço de repetição que informa ao usuário qual o erro e só sai do loop quando ele informa uma entrada válida. A validação das entradas também funciona caso o usuário digite um valor numérico que está fora da faixa de opções e nas entradas que são valores monetários converte a vírgula utilizada no padrão brasileiro para o ponto que é utilizado pelo padrão internacional.

2.3. Desenvolvimento do Sistema

O desenvolvimento do algoritmo se deu da seguinte forma, primeiramente foi elaborado um fluxograma, disponível no Anexo A, com todas as etapas do código. A

construção do código iniciou-se com a declaração de todas variáveis contadoras e acumuladoras que seriam utilizadas ao longo do código, em seguida foi desenvolvido as configurações, configurando as entradas necessárias e suas respectivas validações, além de definir a estética do programa e a paleta de cores. A principal dificuldade nesta etapa foi a validação das entradas de preço, pois no Brasil utiliza-se a vírgula para separar os centavos, já o padrão internacional reconhecido pelo Python utiliza ponto como separador. Por isso foi necessário implementar uma etapa a mais na verificação que substitui a vírgula pelo ponto.

Após isto foi iniciado o desenvolvimento do sistema de vendas, o primeiro desafio foi como fazer com que as vendas permanecessem acontecendo até acabar os ingressos, para isso foi utilizado um laço de repetição While com um condição associada a variável de ingressos restantes. Dentro do loop foram criados os menus de opção que apareceriam na tela e as entradas para cada um desses menus e suas respectivas validações. Em cada um dos menus existem as opções disponíveis e opção de cancelar a venda, que permite o usuário finalizar a repetição e iniciar uma nova, com exceção do menu de escolha de vendedores, onde a opção de finalizar encerra o loop e passa para a exibição de resultado.

A penúltima etapa da repetição do sistema de vendas é o registro da idade, onde o usuário informa a idade do comprador, é feita a validação desta entrada em seguida o valor dessa entrada é incrementado na variável de soma das idades. A última etapa da repetição é a incrementação de todas as variáveis de acordo com o tipo de venda realizada e o decremento da variável de ingressos restantes, que permite o código encerrar quando todos os ingressos são vendidos.

Por fim, as últimas etapas do desenvolvimento do sistema foram: o cálculo dos resultados e a exibição para o usuário. No cálculo dos resultados é realizado todas as operações necessárias utilizando as variáveis de cada categoria, encontrado assim valores como: o total arrecadado, o total arrecadado por cada tipo de ingresso, a quantidade de ingressos emitidos e vendidos. Nesta etapa também foram utilizadas uma série de condições para determinar qual foi o tipo de ingresso mais vendido. A última etapa do sistema também é a mais simples, pois para exibir os resultados, foi necessário apenas utilizar o comando de impressão na tela do valor das variáveis de cada resultado desejado.

Todo o desenvolvimento do projeto foi feito no sistema operacional Windows 11 Home Versão 23H2, na versão 3.12 do Python, utilizando a IDE PyCharm 2024.1, também foi utilizado a ferramenta online Draw.IO para a criação do fluxograma do projeto.

3. Resultados e Discussões

O sistema para este problema foi dividido em quatro partes principais: A configuração inicial, o sistema de vendas, o cálculo dos resultados e a exibição dos resultados. O sistema funciona na seguinte ordem: O usuário insere as informações das configurações iniciais que definem os parâmetros do programa como a quantidade de ingressos disponíveis e o valor do ingresso, em seguida o programa entra no loop do sistema venda, onde são realizadas as escolhas de cada categoria (Tipo de Vendedor, Tipo de Ingresso, Tipo de Meia, Tipo de Venda) e registrado cada venda, quando a

condição do loop for atendia ou a opção de encerrar a venda for selecionada o sistema avança para o cálculo dos resultados e logo em seguida exibe na tela todos os resultados (Quantidade de Ingressos emitidos, Quantidade de Ingressos não emitidos, Quantidade de Inteiras Vendidas, Quantidade de Meias-Entradas Totais, Quantidade de Meias-Entradas Estudante, Quantidade de Meias-Entradas Outros Critérios, Quantidade de Ingresso Preço Especial, Quantidade de Cortesias, Quantidade de Cortesia dos Vendedores, Quantidade de Ingressos vendidos por vendedores comissionados, Valor Total Arrecadados, Valor total de inteiras, Valor total de meias, Valor total de ingresso com preço especial, Maior número de ingressos vendidos.).

Durante os teste realizados o principal erro encontrado foi nas entradas, que caso o usuário digitasse um valor que não fosse inteiro o programa apresentava “ValueError: invalid literal for int() with base 10:”, para solucionar isto foi alterado o tipo de entrada de inteiro para string e feito uma validação. Em todas as entradas do código é feita uma validação que só permite que o programa avance quando o usuário inserir uma entrada válida, caso seja inválida o programa exibe uma mensagem de erro e solicita que a entrada seja inserida novamente. São consideradas entradas inválidas letras (exceto nas perguntas de Sim ou Não), espaços vazios, caracteres especiais e números que estejam fora da faixa de opções disponíveis. Nos casos onde a entrada precisa ser um número, como no caso da idade, é feita a conversão de string para inteiro.

O único erro encontrado nos teste não solucionado ocorre nas entradas de preço, que possuem uma validação específica para a substituição de vírgula por ponto. Neste caso se o usuário utilizar vírgula ou ponto mais de uma vez na entrada como por exemplo: “2.500,50”, o sistema não consegue converter a string para float e apresenta “ValueError: could not convert string to float: '2.500,50’”. Para solucionar este erro seria necessário utilizar a função “.count()” para verificar a quantidade de vírgulas e pontos e caso fosse maior que 1 repetiria o processo de entrada.

4. Conclusão

O sistema desenvolvido atende todos os requisitos mínimos solicitados pelo problema, permitindo a customização, o registro dos dados e a exibição dos mesmos. Além do solicitado, o sistema também possui a validação de todas as entradas e a opção de cancelamento da venda a qualquer momento. Futuramente o programa poderia ser melhorado com a ampliação das opções customizáveis, permitindo que o usuário pode definir melhor o sistema de acordo com cada evento a ser realizado, além da implementação de um sistema de datas, que encerraria o programa no momento desejado depende da quantidade de ingressos restante ou da escolha do usuário.

5. Referências Bibliográficas

PYTHON SOFTWARE FOUNDATION. Python Language Site: Documentation, 2024. Página de documentação. Disponível em: <<https://docs.python.org/3/>>. Acesso em: 19 de abr. de 2024.