Aluno: Henrique Frisso Oliveira e Lorenzo Oliveira De Andrade

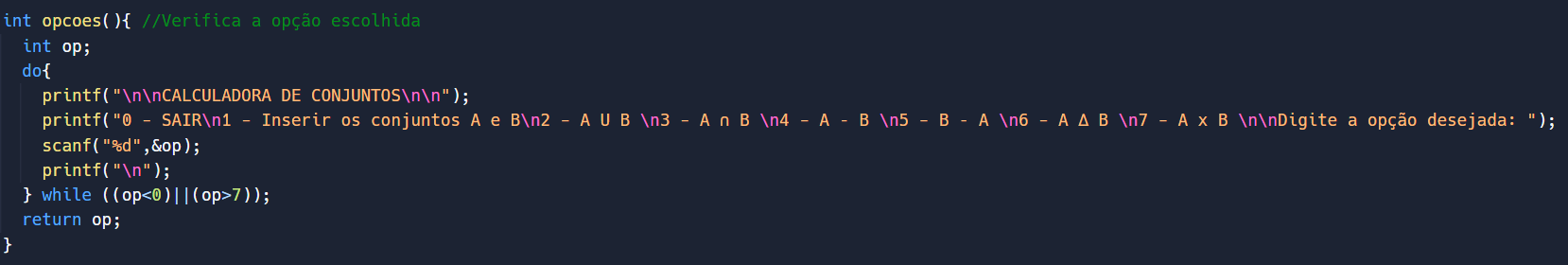
Turma: V06

**Calculadora de Conjuntos**

Para desenvolver uma calculadora de conjuntos vamos precisar de três arrays , dois contendo os conjuntos indicados pelo usuário e um para o conjunto resultante da operação, quando o usuário desejar adicionar conjuntos diferentes apagaremos os conjuntos já existentes e pediremos novamente o tamanho do conjunto e os elementos nele contido, para cada operação armazenaremos o resultado em um vetor criado e ao final da impressão do resultado apagaremos esse vetor para que ele possa ser recriado novamente em outra operação.

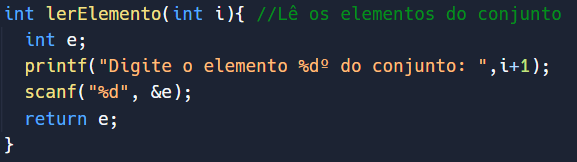


**Função “opcoes”**



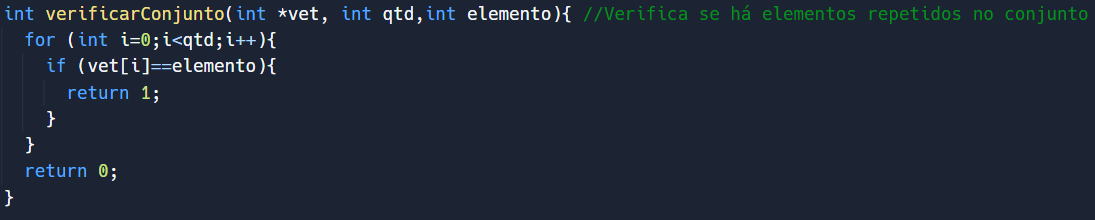
Esta função mostra o index da calculadora, lê a opção escolhida pelo usuário e a retorna.

**Função “lerElemento”**



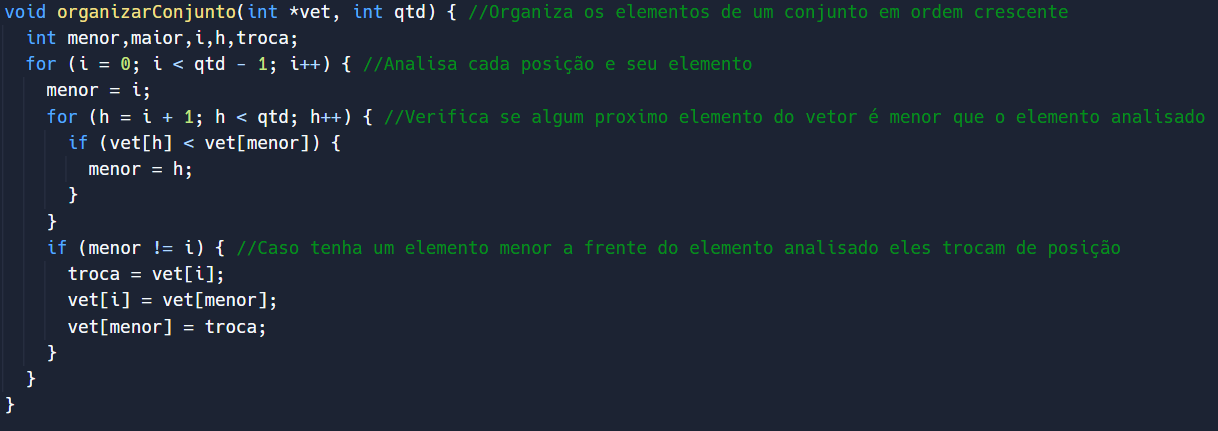
Esta função é utilizada quando o usuário precisa inserir elementos em um conjunto.

**Função “verificarConjunto”**



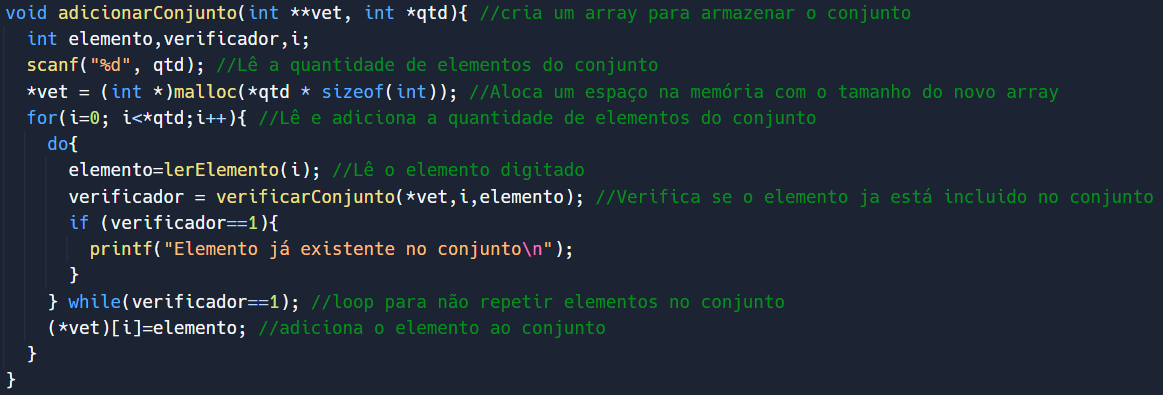
Esta função verifica se um elemento está contido em um conjunto percorrendo o vetor e comparando com o elemento desejado.

**Função “organizarConjunto”**



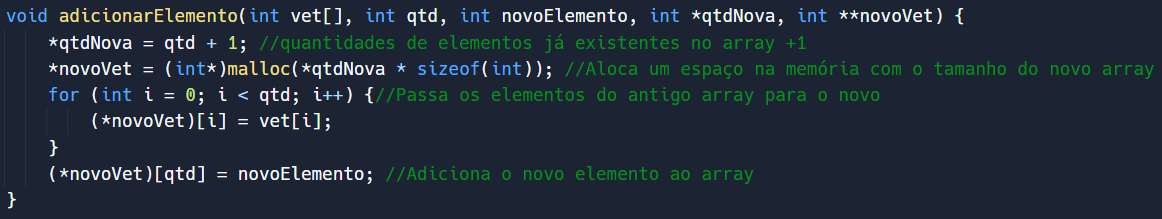
Esta função é utilizada para ordenar conjuntos em ordem crescente, ela recebe o conjunto e compara o primeiro elemento com os demais, caso haja algum elemento seguinte menor que o elemento analisado é feito a troca de lugar, então verifica o próximo elemento com os seguintes e repete até que chegue ao final do vetor.

**Função “adicionarConjunto”**



Esta função é utilizada quando o usuário tem que inserir um conjunto, ela aponta um vetor para um espaço alocado na memória com o tamanho informado pelo usuário, em seguida ela realiza a chamada da função “lerElemento()” e adiciona ao vetor após verificar se o usuário não inseriu elementos repetidos utilizando a função “verificarConjunto()”.

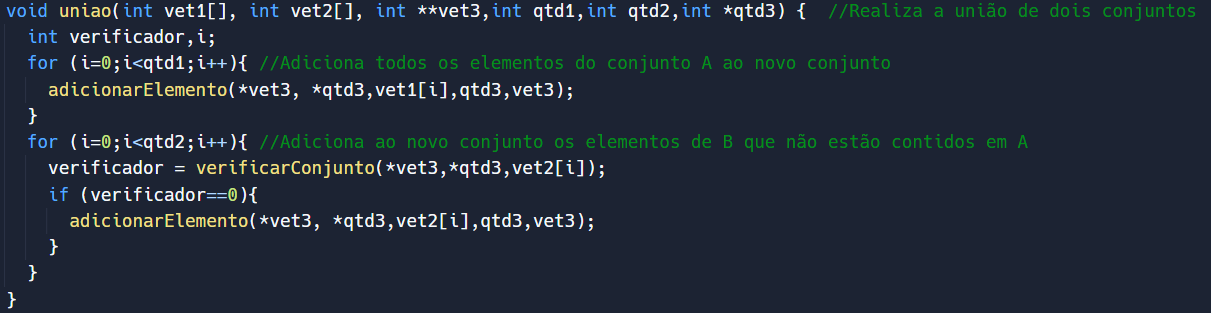
**Função “adicionarElemento”**



Esta função é utilizada para adicionar um elemento ao vetor responsável pelas operações (União,Interseção,Diferença,Diferença Simétrica e Plano Cartesiano), ela recebe o vetor da operação e a quantidade atual de elementos que ele possui, então ele soma um a quantidade, e aloca um espaço na memória para um vetor com a nova quantidade de elementos, depois esse vetor recebe todos os termos que o antigo vetor possuía e adiciona o novo elemento ao vetor.

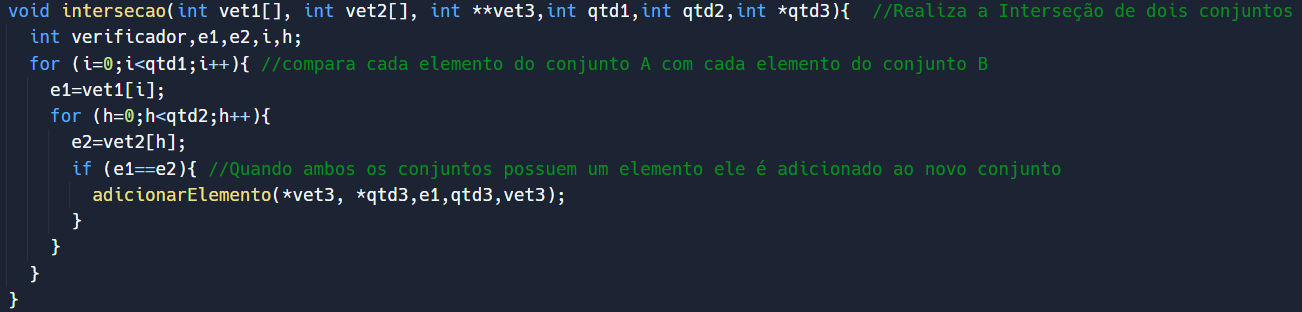
**FUNÇÕES DE OPERAÇÕES**

**Função “uniao”**



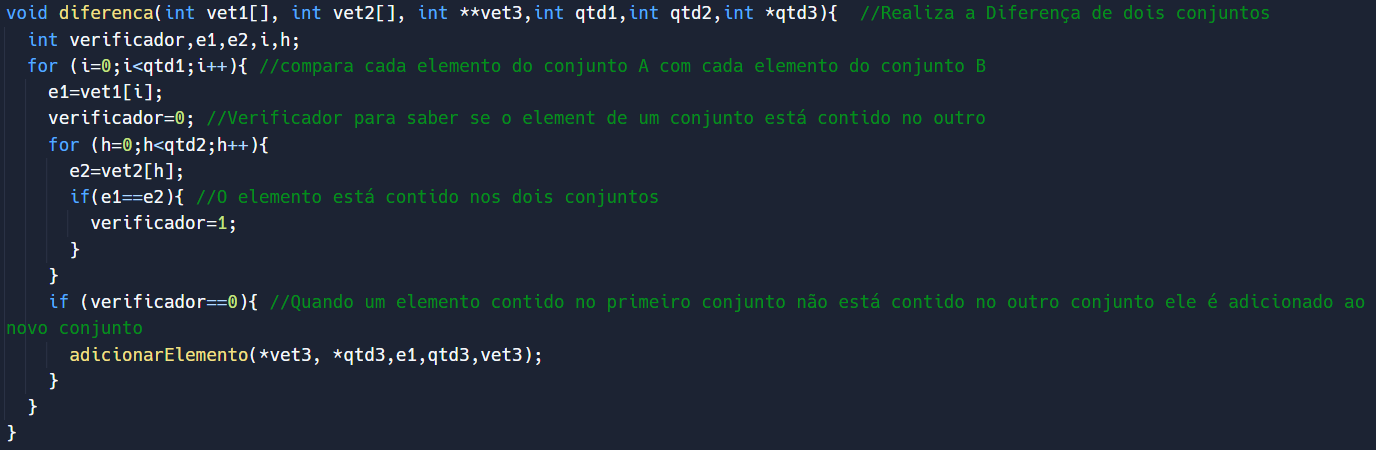
Esta função realiza a operação de união entre dois conjuntos, ela adiciona o primeiro conjunto ao conjunto da operação, e depois verifica todos os elementos do segundo conjunto, caso o elemento não esteja incluso ele é adicionado com a função “adicionarElemento()”.

**Função “intersecao”**



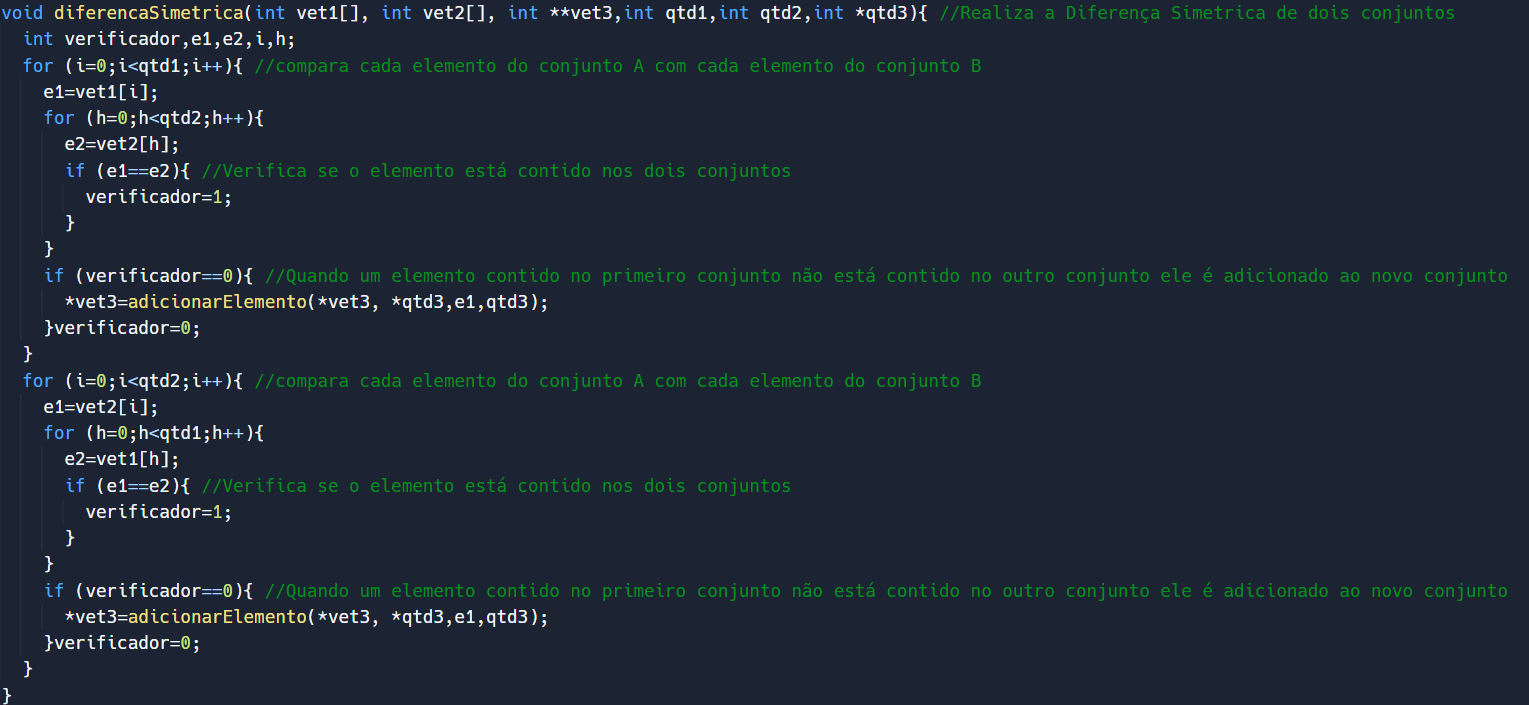
Esta função realiza a operação de interseção entre dois conjuntos, ela percorre ambos os vetores comparando seus elementos, quando ambos os conjuntos possuírem um elemento ele é adicionado ao vetor utilizando a função “adicionarElemento()”.

**Função “diferença”**



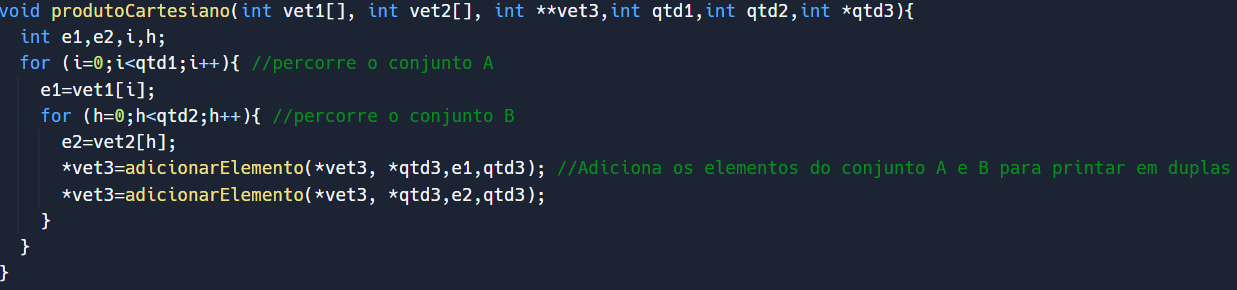
Esta função realiza a operação de diferença entre dois conjuntos, ela percorre ambos os conjuntos comparando seus elementos, e verifica se há elementos iguais, quando há elementos iguais ele altera o valor do verificador para 1 para saber que o número não é único do primeiro conjunto, quando o verificador passa por todos os elementos do segundo conjunto sem ter seu valor alterado ele adiciona o elemento ao vetor que recebe o conjunto diferença através da função “adicionarElemento()”

**Função “diferencaSimetrica”**



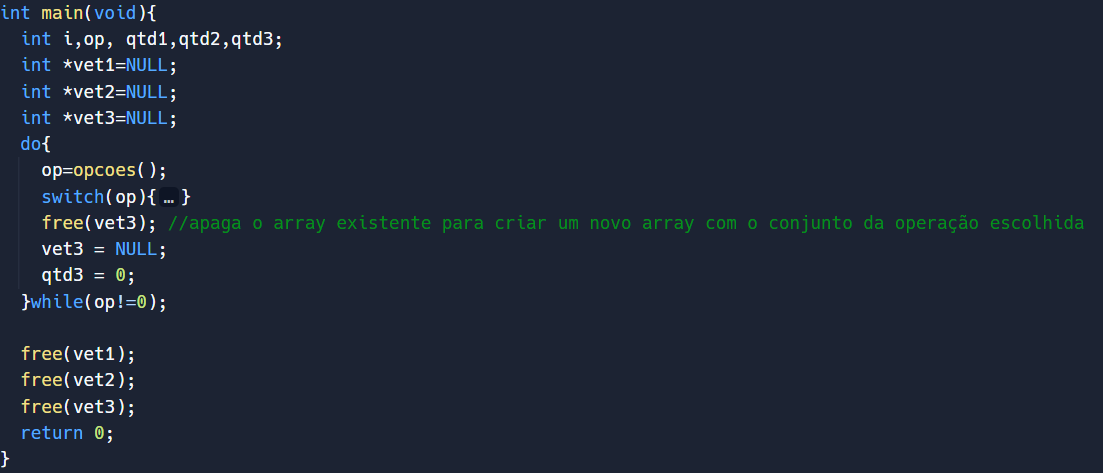
Esta função realiza a operação de diferença simétrica entre dois conjuntos, ela percorre ambos os conjuntos verificando se os elementos de A estão contidos em B e verifica se os elementos de B estão contidos em A, caso não esteja ele adiciona o elemento ao vetor que recebe o conjunto diferença simétrica através da função “adicionarElemento()”

**Função “produtoCartesiano”**



Esta função realiza a operação de plano cartesiano entre dois conjuntos,ela percorre cada conjunto do conjunto A com todos os elementos do conjunto B, adicionando ao conjunto resultante em duplas

**Função main:**



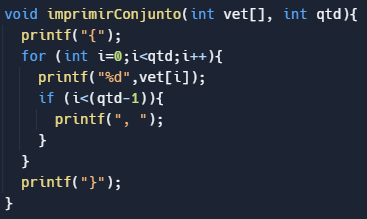
Criação de variáveis, deslocamento e reiniciação do vetor que recebe o conjunto resultante após cada operação, deslocamento de memória no final do código.

“NULL’’ é responsável por deixar o vetor vazio.

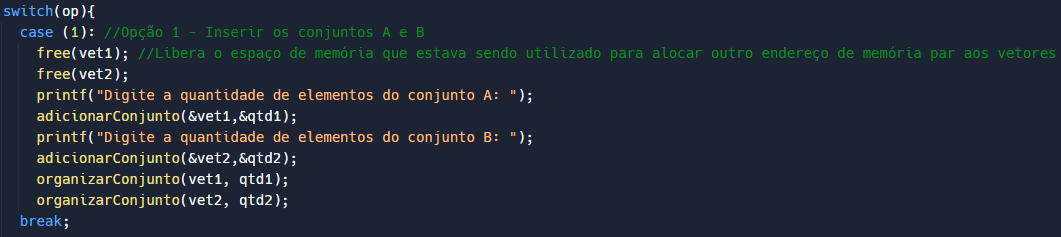
“FREE” é responsável por liberar o espaço alocado na memória.

Impressão de resultados:

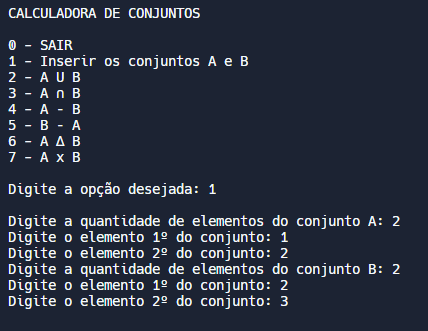
**Função “imprimirConjunto”**

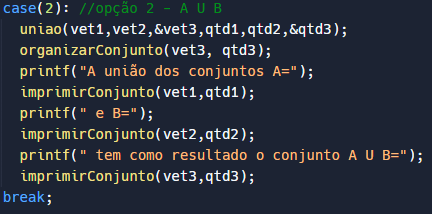


Essa função é responsável por realizar a impressão do conjunto, percorrendo-o por completo.

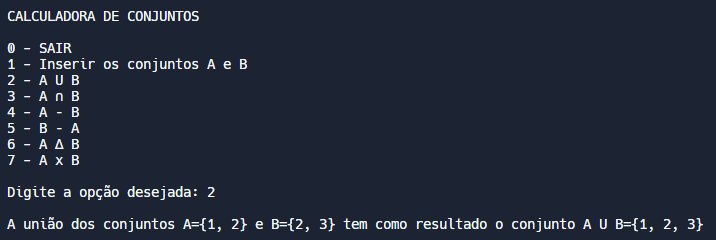


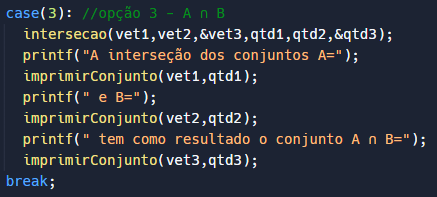
Neste parte do código, além e imprimir e recolher a quantidade de elementos, faz a chamada das funções responsáveis por adicionar e organizar os elementos dentro dos conjuntos.



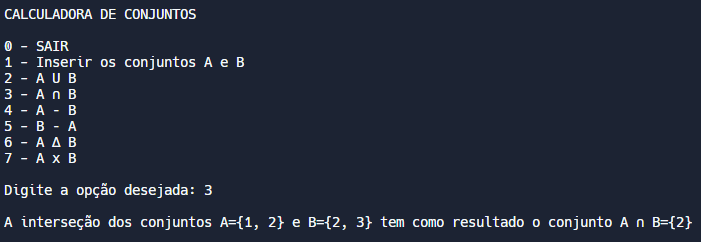


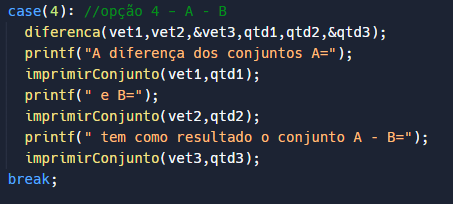
Case 2, é responsável por realizar a operação de União entre o Conjunto A e o Conjunto B e exibi-la, retornando ao menu principal, ao final da operação.



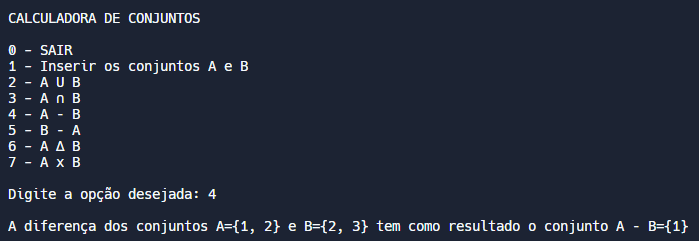


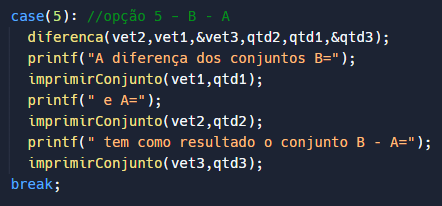
Case 3, é responsável por realizar a operação de Intersecção entre os Conjuntos A e o Conjunto B e exibi-la, sendo interrompido ao final da operação, retornando ao menu principal.



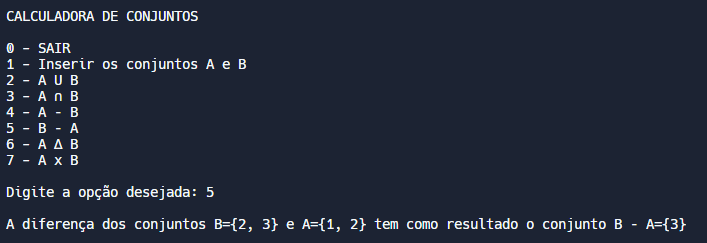


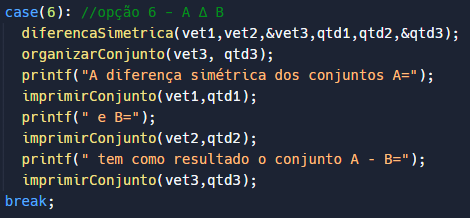
Case 4, é responsável por realizar a operação Diferença entre os Conjuntos A e B e exibi-la, onde é utilizado como base as mesmas funções utilizadas no case 4, porém, a diferença é o conjunto usado como base, sendo interrompido ao final, retornando ao menu principal.



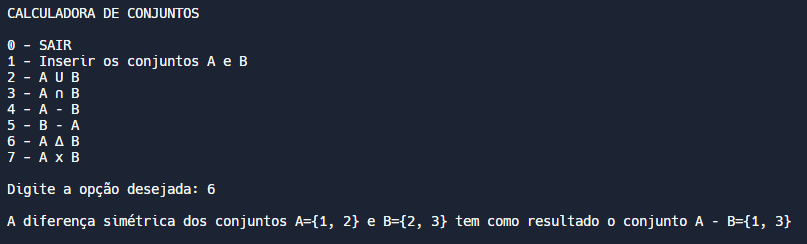


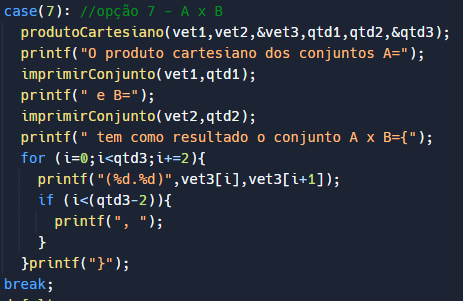
Case 5, é responsável por realizar a operação da Diferença entre os Conjuntos A e Conjunto B, onde é utilizado como base as mesmas funções utilizadas no case 3, porém, a diferença é o conjunto usado como base, sendo interrompido ao final, retornando ao menu principal.





Case 6, é responsável por realizar a operação Diferença Simétrica entre os Conjunto A e o Conjunto B e exibi-lá , sendo interrompido ao final da operação, logo após, é feito exibido novamente o menu de opções disponíveis para o usuário escolher.





Case 7, é responsável por realizar a operação do Plano Cartesiano entre os Conjuntos A e o Conjunto B e exibi-la, sendo interrompido ao final da operação, logo após, é feito exibido novamente o menu de opções disponíveis para o usuário escolher.

