Alunos: Thiago Rangel da Silva, Pedro Augusto Figueiredo Gonzaga

Preencha para cada uma dos paradigmas abaixo

* Explique como cada solução funciona (quais funções, métodos fazem o que)
* Quais foram as dificuldades e facilidades de cada um dos paradigmas

Estruturado

|  |
| --- |
| Na programação estruturada devido as limitações que foram colocadas no trabalho. Para tratar o texto retirando caracteres especiais resolvi fazer um dicionário com todos os caracteres especiais da língua portuguesa com o seu respectivo valor sem o caracter especial. Recebo o texto e transformo num objeto List(), assim separando o texto numa lista de caracteres separados e tratando eles de um a um, se for caracter maiúsculo eu pego o valor ASCII dele com a função ord() e adiciono +32 porque percebi que toda letra maiúscula a sua minúscula é seu valor ASCII + 32.  Depois de ter os caracteres tratados utilizo o append para um outro objeto List(), assim dou um join com essa List() a uma string vazia, fazendo a List() voltar a ser uma string de palavras. Utilizo o split() com o separador padrão sendo o espaço e faço um for percorrendo a lista de palavras jogando no LIWC, após isso faço um contador para cada emoção que foi pedida no exercício e faço as porcentagens de posemo e negemo. |

Orientação a objetos

|  |
| --- |
|  |

Funcional

|  |
| --- |
| Na programação funcional decidi usar Node.JS devido a saber mais sobre utilizar as funções definidas. Pelo que eu entendi do exercício tudo deveria ser constante, sem loops e usando recursão. Tinha que também utilizar map, filter e reduce a vontade ao menos 1 vez, na minha interpretação não era necessário utilizar os 3 pelo menos 1 vez e sim pelo menos 1 delas 1 vez. Abri o arquivo txt usando o readFile e salvei o texto numa constante, após isso realizei o split do texto nos espaços virando uma lista de palavras. Após isso nessa lista de palavras utilzei o map para colocar todas em minúsculo e utilizando uma expressão regular para limpar caracteres especiais e pontuações da string salvando este resultado em uma nova constante.  Após isso, criei uma função recursiva chamada contarSensacoes que recebe como parâmetros o array de palavras tratadas, um índice, e a contagem, esta função passa por cada palavra do array de palavras e conta as sensações pedidas no exercício salvando em contagem. Assim que o índice for do tamanho da length do array de palavras tratadas a função retorna o objeto contagem.  Por fim criei uma const chamada ContagemInicial com os valores setados = 0 para cada sensação e passei por parâmetro na chamada da função criada junto com o array de palavras e o índice = 0. |