

Projeto Prático 02

Parte 1: Vetores

Clayton S. Rocha, Lucas G. Santana, Mateus M. Leal.

2194201 - 2088924 - 2126702

Departamento de Informática – Universidade Tecnológica Federal do Paraná (UTFPR)
Av. Sete de Setembro, 3165 – 80280-901 – Curitiba – PR – Brasil

Bacharelado em Sistemas de Informação – Universidade Tecnológica Federal do Paraná
Curitiba, PR.

Resumo. Relatório sobre o Projeto prático 02, parte 01:Vetores. O mesmo contém detalhes de como foram feitas as resoluções para os problemas apresentados, descrevendo o funcionamento das funções utilizadas para elaboração do projeto.

1. Brainstorming

No início houveram muitas dificuldades para entender a proposta, entender sua estrutura, como funciona a manipulação do áudio, sobre as funções do módulo wavfile, e até mesmo a criação de um projeto no CodeBlocks.

Inicializamos o trabalho em uma das aulas de atendimento da professora Leyza Baldo Dorini, onde conseguimos sanar quase todas as duvidas existentes em relação a proposta do projeto.

Logo após o esclarecimento de todas a dúvidas, iniciamos o processo de criação com base no passo a passo disponibilizado em PDF no moodle. Criamos as pastas necessárias, criamos um projeto no CodeBlocks, e utilizamos os arquivos também cedidos no moodle pelos orientadores para poder prosseguir com projeto.

A primeira função, mudaGanho implementada por nós foi de certa forma rápida, pois contamos com o auxilio e explicações da professora Leyza para realizar a implementação necessária. Utilizamos um “for” do tamanho `n_amostras`, para percorrer todo vetor, e atualizando o vetor com a soma dos dados iniciais mais o ganho. Fazendo com que o ganho do áudio seja alterado de acordo com o sinal.

`ruidoPeriodico` foi a segunda função implementada, percorrendo todo o vetor de tamanho `x` em um intervalo `y`, adicionamos um ruído na posição 0 com seu valor 1, e assim vai intercalando de 1 e -1, em um intervalo passado por parâmetro. Assim fazendo com que seu áudio final saia com uma espécie de estalo.

A terceira função proposta utiliza a anterior como base, já que na segunda tivemos de adicionar o ruído, nessa atual função iremos criar um algoritmo para remover os possíveis ruídos, percorrendo todo vetor, e retirando sua mediana de 3 amostras a cada “memoria alocada”, exemplo(`vet[5]{1,2,3,4,5}` `vet[1] = mediana;` `mediana = valor médio entre vet[0], vet[1] e vet[2]`, e assim por diante.) assim nivelando todo o áudio, pois tirando a mediana de 3 valores, acabamos removendo todos os extremos locais, sempre pegando o ponto médio dos valores do vetor, assim nunca sobrando a possibilidade da obtenção de ruídos no áudio.

Já a quarta e última função começamos a fazer durante a aula de atendimento, porém foi a que demandou mais tempo e esforço para compreensão, pois não era apenas produzir o algoritmo para simular a sub smostragem, teríamos de também fazer a chamada da função, onde nos confundimos bastante. Nos reunimos na biblioteca para finalizar a ultima função restante, porém nunca conseguíamos o resultado desejado, pois não tínhamos um áudio para poder fazer um comparativo para saber se o resultado obtido era o esperado. Então resolvemos deixar para resolver outro dia através de vídeo conferência.

Através de uma nova reunião em uma vídeo conferencia testamos diversos tipos de algoritmos, varias opções logicas, e sempre obtendo resultados diferentes, mas nunca sabíamos se o efeito que foi obtido era o desejado, até que procuramos ajuda para conseguir resolver o mesmo, foi onde conseguimos sanar nossas duvidas e por consequênciia conseguir completar o trabalho.

Apesar de parecer um projeto muito complexo no início, após sua compreensão de proposta tudo acaba sendo facilitado, e através de maneiras lógicas consegue realizar de maneira eficaz a conclusão, todo auxílio e passo a passo para realizar a criação de funções, projetos entre outros foram cedidos em arquivos no moodle, o que facilitou e possibilitou a entrega do mesmo.