Questão 4 – Relatório da Implementação

Na questão 1 implementamos a estrutura HASH, sem tratamento de colisões, em três funções de mapeamento.

Para a função dobra iniciamos com chave e declaramos int num_bits, que é igual a chave, depois pegamos as duas partes, ou seja, parte 1 e parte 2 e atribuímos para chave >> num_bits. Então pegamos a parte 2 e atribuímos para chave & TAM que é o tamanho definido no define TAM 7. Logo retornamos à função do tipo inteiro com parte 1 ^ parte 2 e finalizamos a função dobra.

Já na função multiplicação já temos nossa chave declarada e uma variável a = 0.6180339887 que já tem um valor definido para usar. Então criamos uma variável float chamada value que atribui para chave multiplicado com a variável a. Logo value atribuir para value menos (int) value transformando a value em um inteiro porque era tipo float. Desse modo retornamos essa função com (int) nos parênteses e (TAM * value) multiplicamos esse TAM e value.

Na função divisão também criamos uma chave, mas do tipo inteiro e está retornando a chave pelo modulo do TAM que é o tamanho da tabela para achar a posição do valor inserido.

No resultado final aparecem três opções no menu, ao escolher inserir o usuário pode colocar um valor desejado, em seguida temos o imprimir, que atribui o valor colocado anteriormente na lista, por fim tem o sair, que encerra o programa.

Já na questão 2, nós implementamos a estrutura HASH com tratamento de colisões.

Na letra A da 2 definimos o tamanho da tabela com o TAM. E com isso criamos duas struct, uma chamada No e a outra chamada Lista, depois criamos uma função inicializaLista que é usada para organizar a lista da tabela HASH e ainda temos a funçaodiv que é usada para calcular a posição de cada elemento na tabela. Logo temos a função de inseriLista que inseri numa lista encadeada. Então a função busca na lista seu retorno e um inteiro que ela usa para buscar elementos numa tabela de lista encadeada e a função void usada para imprimir a tabela de uma lista encadeada. Ainda termos a função remover usada para remover um numero da tabela de uma lista encadeada.

Depois usamos o inserir com o vetor inteiro, passamos a Lista para a struct que declaramos no começo do código e chamamos a função inserir na lista dentro dessa função. E então temos a função inserir na tabela de vetor, usamos a Lista em vez de int e chamamos a função inicializarLista dentro dessa função. A função busca, também de um vetor passando Lista, tem retorno inteiro e tem a função buscaLista dentro dela para facilitar a busca numa lista encadeada. Com isso a função imprimir que também tem o imprimirLista dentro dela para imprimir em uma lista encadeada e a função apagar que tem a mesma função das anteriores que é usada para apagar valores da lista encadeada, além de ter dentro dela a função remover da lista encadeada. Assim termos a função menu que vai mostrar todas as operações que o código pode fazer e temos um switch para

facilitar o menu, com cada case uma opção de escolha, nós temos 5 opções que são inserir, busca, mostra, remover e sair do programa.

Já na letra B da 2 temos o início do código e o TAM 7 que é o tamanho da tabela. E a função incilizarTabela que é usada para iniciar a tabela em um vetor, a função div, usada para achar a posição de um valor inerido na tabela pelo seu resto da divisão. Com isso tivemos que inserir os elementos na tabela em forma de vetor. O imprimir foi inserido na tabela usando vetor. Por fim a última função que o menu mostra ao usuário pode escolher entre 3 opções inserir, mostra ou sair que fecha o programa. Depois vem o swict para aplica as opções do menu e passar as funções inserir e mostra.

Por último. na questão C da 2 vemos uma lista encadeada com o TAM 7 com a função inicializarLista que inicializa a lista encadeada, inserirLista que insere os valores na lista encadeada, imprimirLista que imprimi os valores que foram inseridos na lista encadeada. Com isso temos o iniciliazartabela, que inicializa o vetor com tipo lista em vez de inteiro, a função div que calcula a posição do valor inserido, que inseri num vetor, mas ele tem uma função inserirLista dentro dele, assim inserindo em uma lista encadeada. Imprimir, com vetor que também usa Lista em vez de inteiro e tem a função imprimirLista dentro dela também para imprimir numa lista encadeada e por último a função menu usada para mostra o usuário as opções como inseri, imprimir e sair do programa. Assim temos o main que termos o switch com as opções do menu passando as funções que forma citada no menu.

Na questão 3, apresentamos exemplos das funções de HASH apresentadas na questão 1.

Dessa forma, temos 3 tipos que definem o TAM que é usado para inserir e mostrar a divisão, o TAMM usado para a multiplicação e o TAMD para a dobra. Temos várias funções de inicialização termos a incilaizarDiv que inicializa a parte da divisão, a de inicializaTabelaMult que inicializa a tabela de multiplicação e a de inicializarTabelaDobra que inicializa a tabela dobra. Assim termos 3 funções importante para achar a posição de um valor inserido na tabela HASH que são a funçaodiv, funçaoDobra e funçaoMult.

Então colocamos a inserção por vetor dessas três funções que são inserirMult, inserirDobra e inserirDiv, em seguida usamos imprimir que são imprimirMult, imprimiDobra e imprimirDiv. Para finaliza as funções termos 3 menus que são usados para mostrar as opções de escolha para o usuário do menu1, menu2 e menu3 com isso usamos o switch para escolhermos qual das funções, inserir ou mostrar, com isso o usuário pode escolher qual função quer inserir e mostrar a tabela.