

Trabalho de PDS II - Implementação de Tipo Abstratos de Dados

Agenda de Compromissos

Alunos:

Maria Teresa Menezes Costa 2018092892

Otávio Augusto Soares Oliveira 2018072204

Professor: Helton Fábio de Matos

Problema

O exercício implica em implementar um programa em C++, sem a utilização de TAD's já definidas em bibliotecas como a STL, para administrar uma Agenda de compromissos. A Agenda deve conter o mês, o conjunto de dias do mês e uma lista de compromissos para cada dia do mês. Cada entrada deve armazenar um apontador para o primeiro compromisso da lista, caso não exista, deve ser armazenado o apontador nulo. Já o compromisso deve possuir pelo menos três campos, sendo eles a hora do compromisso, um campo para a descrição e o terceiro para armazenar o apontador de encadeamento dos compromissos do dia.

Solução

Para solucionar o problema, fizemos uma matriz de listas encadeadas. A matriz refere-se aos 12 meses do ano e aos 31 dias (máximo) de cada mês. Dessa forma, cada dia do ano possui uma lista encadeada de compromissos, possibilitando ao usuário inserir a quantidade de compromissos que quiser por dia.

```
typedef struct Nodo{
    Nodo();
    ~Nodo();
    struct Nodo *prox;
    int isEmpty;
    string compromisso;
}Nodo;

typedef struct Lista{
    Lista();
    ~Lista();
    Nodo *l[12][31];
}Lista;
```

As opções mostradas ao usuário para manipular os dados na agenda são Abrir, Listar, Inserir, Remover, Verificar e Fechar.

1. Abrir a agenda
2. Inserir compromisso
3. Remover compromisso
4. Listar compromissos
5. Verificar se existe compromisso agendado
6. Salvar e fechar

Os dados necessários para que um compromisso seja inserido são o mês, o dia, o horário e o compromisso.

Ao escolher a opção remover, o usuário precisa entrar com o mês, o dia e o horário do compromisso que deseja excluir:

```
Insira o mes
04
Insira o dia
25
Insira a hora e o minuto <Formato 0000>
1400
Compromisso Removido
```

Caso não exista um compromisso no mês/dia/horário informado, o programa indica para o usuário que ele não existe.

Ao listar os compromissos salvos na agenda, o programa imprime ordenado os meses e os dias. Além disso, o programa também ordena por horário os compromissos de um mesmo dia:

```
-----
##25 de Abril##
-----
14:00 prova de PDS II
-----
##26 de Abril##
-----
15:20 Relatorio de PDS II
-----
##28 de Abril##
-----
12:00 Entrega do trabalho de PDS II
-----
##1 de Maio##
-----
11:00 Almoçar com Maria Tereza
15:20 Buscar exame
```

Ao escolher a opção 5, o programa verifica se existe no dia e no mês desejado algum compromisso. Caso exista, ele pesquisa pelos compromissos no horário indicado pelo usuário. Caso não existam compromissos no dia/mês/horário indicado, o programa também indica este fato ao usuário.

```
Insira o mes
04
Insira o dia
25
Insira a hora sem os minutos
14
Voce tem compromisso(s) agendado para esse horario:
14:00 prova de PDS II
```

Por fim, para salvar e fechar, o programa grava no arquivo todas as informações inseridas pelo usuário.

```
void escreverParaArquivo(ofstream &file, Lista *lista){
    Nodo *aux;
    string dia, mes;
    for(int i = 0; i < 12; i++){
        for(int j = 0; j < 31; j++){
            aux = lista->l[i][j];
            while(aux->isEmpty == 0){
                if(i+1 < 10)
                    mes = "0" + to_string(i+1);
                else
                    mes = to_string(i+1);

                if(j+1 < 10)
                    dia = "0" + to_string(j+1);
                else
                    dia = to_string(j+1);

                file << mes << dia << aux->compromisso << "\n";
                aux = aux->prox;
            }
        }
    }
}
```