**UNIVERSIDADE DE FRANCA – UNIFRAN**

FABIO TOLEDO BONEMER DE SALVI

**ROTINA DE ATIVIDADE FÍSICA**

Experiência Prática II

ARARAQUARA – SÃO PAULO

2025

FABIO TOLEDO BONEMER DE SALVI

**ROTINA DE ATIVIDADE FÍSICA**

Experiência Prática I

Trabalho apresentado ao curso de Superior de Tecnologia em Análise e Desenvolvimento de Sistemas da Universidade de Franca – UNIFRAN, como requisito para obtenção de nota de Experiência Prática II na disciplina Algoritmos e Pensamento Computacional.

ARARAQUARA – SÃO PAULO

2025

# Introdução

O presente trabalho tem como objetivo desenvolver um programa funcional em linguagem C, aplicando corretamente algumas estruturas de repetição (como *while*, *for* ou *do-while*) e alguma estrutura de decisão (como *if*, *else* ou *switch*).

A proposta visa exercitar a lógica e o pensamento computacional por meio da resolução de problemas reais, explorando a criatividade e a interação com o usuário.

O problema foi implementado em linguagem de programação C utilizando o ambiente de desenvolvimento *Visual Studio Code* ([MICROSOFT CORPORATION, 2025](https://code.visualstudio.com/)) em conjunto com as ferramentas de desenvolvimento e bibliotecas fornecidas pelo projeto *Minimalist GNU for Windows* ([MINGW-W64 PROJECT, 2025](https://www.mingw-w64.org/)) que contém ferramentas e bibliotecas que possibilitam o desenvolvimento de aplicações de forma nativa em ambiente *Windows*.

A instalação e configuração do ambiente de desenvolvimento foi realizada seguindo os passos descritos em [MICROSOFT (2025](https://code.visualstudio.com/docs/cpp/config-mingw)) onde é descrito como configurar o compilador GCC C++ (g++) e depurador GDB.

O diretório github deste projeto está disponível em [Salvi (2025)](https://github.com/engfabiodesalvi/physical-activity-recording-system.git).

# Descrição do Problema

O objetivo é desenvolver um software em linguagem de programação C que registre as rotinas diárias de atividades físicas e possibilite a sua visualização, permitindo ao usuário criar, armazenar e visualizar suas rotinas diárias de atividades físicas através da interação com o software realizada por exibições de mensagens objetivas em tela e envio de comandos.

A solução proposta permite a usuário visualizar as atividades físicas diárias, cadastrar uma atividade física diária, cadastrar novas categorias de atividades físicas e cadastrar novas atividades físicas.

## Requisitos que foram atendidos

O programa desenvolvido atende aos seguintes itens:

### Possui estrutura de repetição (laço) *while*, *for* e *do-while*;

### Contém estrutura de decisão (condicional) *if/else/else if* ou *switch*;

### Utiliza variáveis com nomes legíveis e significativos, que facilitem a compreensão do código;

### As mensagens e saídas para o usuário são claras e objetivas;

### O código está bem indentado e comentado adequadamente, explicando trechos importantes e a lógica usada;

### Permite que o usuário realize múltiplas operações até optar por sair do programa (*loop* principal).

### Utiliza funções e comandos básicos como *printf*, *scanf*, *if/else*, *switch*, *while*, *for*, entre outros.

### O programa foi testado em diferentes cenários e entradas de dados, funcionando corretamente.

## Organização do Código

### Apresentação do sistema: uma mensagem inicial explica o que programa faz;

### Menu de opções: implementado dentro de um *loop while* e permiti repetição das ações;

### Tratamento das opções: uso de *if*/*else* ou *switch* no controle das funcionalidades conforme a escolha do usuário;

### Validação das entradas: garante que o usuário informe valores válidos e trata entradas incorretas com mensagens amigáveis;

### Mensagens de encerramento: ao sair do programa, apresenta uma mensagem de agradecimento;

# Código do programa em linguagem C

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <ctype.h> // For isspace()

#include <Windows.h> // SetConsoleOutputCP()

// Declarando as atividades físicas

// até 10 categorias

// até 10 subcategorias

// max 50 caracteres por item

// Número máximo de cadastro de categorias

const int MAX\_CATEGORIA = 10;

// Número máximo de cadastro de atividades físicas por categoria

const int MAX\_ATIVIADE\_FISICA = 10;

// Número máximo de caracteres para nomear uma categoria ou uma atividade física

const int MAX\_CARACTERES = 50;

// Número máximo de registros de atividades físicas

const int MAX\_REGISTRO\_ATIVIDADE = 50;

// protótipos das funções

void apresentacaoSistema();

void menuInicio();

void menuFim();

void menuPrincipal();

void menuVisualizarAtividadesFisicas(char diasDaSemana[7][15]);

void menuVisualizarAtividadesFisicasDia(int diaSelecionado, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroCategoria, int atividadesFisicasSelecinadas[50][3], int numeroAtividadesFisicasSelecionadas);

void menuRegistrarAtividadeFisica(char diasDaSemana[7][15]);

void menuRegistrarAtividadeFisicaDia(int diaSelecionado, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroCategorias);

void menuRegistrarAtividadeFisicaDiaCategoria(int diaSelecionado, int categoriaSelecionada, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroAtividadesFisicas);

void menuRegistrarAtividadeFisicaDiaNovaCategoria(int diaSelecionado, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroCategorias);

void menuRegistrarAtividadeFisicaDiaNovaAtividadeFisica(int diaSelecionado, int categoriaSelecionada, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroCategorias);

int validaEntradaNumerica(int opcao, int opcaoMinMax[2], int returnOpcao);

int validaEntradaCaracters(char \*caracteres, char \*returnNovoCaracteres);

void mensagemEncerramento();

void trim(char \*s);

int main() {

    SetConsoleOutputCP( CP\_UTF8 ); //  mostrar caracteres acentuados

    // Definindo as variáveis de manipulação dos menus

    int menu = 0, returnMenu = 0;

    int sairMenu = 0;

    // Definindo as variáveis de manipulação das opções do menu

    int opcao = 0, returnOpcao = 0;

    // Faixa de opções do menu

    int opcaoMinMax[2] = {0,0};

    // Categoria digitada pelo usuário

    char novaCategoria[50] = {""};

    // Array para armazenar o retorno da função fgets() [Erro -> NULL | Success -> aponta para a variável utilizada como buffer]

    char \*returnNovaCategoria;

    // Nova atividade física digitada pelo usuário

    char novaAtividadeFisica[50] = {""};

    // Array para armazenar o retorno da função fgets() [Erro -> NULL | Success -> aponta para a variável utilizada como buffer]

    char \*returnNovaAtividadeFisica;

    // Opções elecionadas pelo usuário

    // Domingo a Sabado - 1 a 7 / Todos os dias - 8

    int diaSelecionado = 0;

    // Categoria Selecionada

    int categoriaSelecionada = 0;

    // Atividade Selecionada

    int atividadeSelecionada = 0;

    // Array das atividades físicas selecionadas

    // {dia\_semana, categoria, atividade\_fisica}

    // dia\_semana -> 0 a 6

    // categoria -> linha do array atividadesFisicas

    // atividade\_fisica -> coluna do array atividadesFisicas

    // array nulu

    int atividadesFisicasSelecinadas[50][3] = {0};

    // Número de atividades físicas inseridas no array atividadesFisicasSelecinadas

    int numeroAtividadesFisicasSelecionadas = 0;

    // Array de exemplo com 11 elementos

    // int atividadesFisicasSelecinadas[50][3] = {

    //     {0, 0, 1}, {1, 1, 1}, {2, 2, 1}, {3, 3, 1}, {4, 4, 1},

    //     {5, 3, 3}, {6, 2, 4}, {1, 3, 2}, {4, 4, 2}, {6, 2, 2},

    //     {6, 3, 2}

    // };

    // Declarando os dias da semana

    char diasDaSemana[7][15] = {

        "Domingo", "Segunda-Feira", "Terça-Feira", "Quarta-Feira",

        "Quinta-Feira", "Sexta-Feira", "Sábado"

    };

    // Número de categrias cadastradas

    int numeroCategorias;

    // Número de atividades físicas cadastradas

    int numeroAtividadesFisicas;

    // {"Categoria",

    //  "Atividade1", "Atividade2", "Atividade3", "Atividade4", "Atividade5", "Atividade6", "Atividade7", "Atividade8", "Atividade9", "Atividade10"

    // }

    // Atividades físicas cadastradas

    char atividadesFisicas[10][11][50] =  {

        {

            "Cardiorespiratórios",

            "Corrida / Trote", "Caminhada", "Ciclismo", "Pular corda", "Natação", "Escalada indoor / step", "", "", "", ""

        },

        {

            "Força / Musculação",

            "Flexões (Push-ups)", "Agachamentos (Squats)", "Afundos (Lunges)", "Levantamento terra (Deadlift)", "Supino (Bench press)", "Remada (Row)", "Prancha (Plank) - para core", "", "", ""

        },

        {

            "Exercícios de Flexibilidade / Alongamento",

            "Alongamento de pernas", "Alongamento de braços e ombros", "Yoga básico", "Pilates", "", "", "", "", "", ""

        },

        {

            "Exercícios Funcionais / HIIT",

            "Burpees", "Mountain climbers", "Jump squats", "Kettlebell swings (se houver equipamento)", "Exercícios com TRX (suspensão)", "", "", "", "", ""

        },

        {

            "Atividades de Equilíbrio / Core",

            "Prancha lateral", "Ponte (Glute bridge)", "Abdominais (Crunches, Bicycle crunch)", "Exercícios com bola suíça", "", "", "", "", "", ""

        },

        {"", "", "", "", "", "", "", "", "", "", ""},

        {"", "", "", "", "", "", "", "", "", "", ""},

        {"", "", "", "", "", "", "", "", "", "", ""},

        {"", "", "", "", "", "", "", "", "", "", ""},

        {"", "", "", "", "", "", "", "", "", "", ""}

    };

    // Mensagem inicial

    apresentacaoSistema();

    // Menu 0 - Menu principal

    menu = 0;

    sairMenu = 0;

    // Loop dos menus

    do {

        if (menu == 0) {

            // Menu para a opção 0

            // Primeiro menu:

            // Menu principal - "Menu de opções"

            // Loop que garante uma entrada numérica válida

            do {

                // Menu principal -> "Menu de opções"

                menuInicio();

                menuPrincipal();

                // Opcao selecionada pelo usuario

                printf("Digite a opção desejada: ");

                returnOpcao = scanf("%d", &opcao);

                menuFim();

                // Opções do menu

                opcaoMinMax[0] = 1;

                opcaoMinMax[1] = 3;

                // Limpando o buffer de entrada!!!!

                while(getchar() != '\n');

            // Valida a entrada numerica antes de prosseguir

            } while (!validaEntradaNumerica(opcao, opcaoMinMax, returnOpcao));

            // Analisa a opção selecionada

            if (opcao >= 1 && opcao <= 2) {

                menu = opcao;

            } else {

                // Sair do programa

                sairMenu = 1;

            }

        } else if ( menu == 1) {

            // Selecionar o dia da semana ou semana inteira

            // Menu para a opção 1

            // Passos para chegar a este menu:

            // "Serviços disponíveis" -> "1 - Visualizar as atividades físicas"

            // Loop que garante uma entrada numérica válida

            do {

                // Menu "Visualizar as atividades físicas"

                // O usuário deve escolher um dia da semana

                menuInicio();

                menuVisualizarAtividadesFisicas(diasDaSemana);

                // Opcao selecionada pelo usuario

                printf("Entre com a opção: ");

                returnOpcao = scanf("%d", &opcao);

                menuFim();

                // Opções do menu

                opcaoMinMax[0] = 1;

                opcaoMinMax[1] = 9;

                // Limpando o buffer de entrada!!!!

                while(getchar() != '\n');

            // Valida a entrada numerica antes de prosseguir

            } while (!validaEntradaNumerica(opcao, opcaoMinMax, returnOpcao));

            // Verifica qual opção do menu (dia da semana) foi escolhida

            if (opcao >= 1 && opcao <= 8) {

                    // Domingo a Sabado - 1 a 7 / Todos os dias - 8

                    diaSelecionado = opcao;

                    // Primeiro dígito (1) - "Visualizar as atividades físicas"

                    // Segundo dígito (opcao) - Identifica o dia da semana ou todos os dias

                    menu = opcao + 10;

            } else {

                    // Opcao 9 - Voltar

                    // Volta para o menu principal

                    menu = 0;

            }

        } else if (menu >= 11 && menu <= 18) {

            // Mostra as atividades física do dia selecinado ou da semana inteira

            // Mostra o "Top Day"

            // Menu para as opçoes de 11 a 18

            // Passos para chegar a este menu:

            // "Serviços disponíveis" -> "1 - Visualizar as atividades físicas"

            // "Visualizar as atividades físicas de qual(quais) dia(s)?" -> "1 - Domingo" a "8 - Semana inteira"

            // Mostra na tela as atividades físicas para o/os dias selecionados

            // Primeiro dígito [1] - Visualizar as atividades físicas

            // Segundo dígito [opcao] - Identifica o dia da semana [1-7] ou todos os dias [8]

            // Loop que garante uma entrada numérica válida

            do {

                // Menu "Vizualizar atividades físicas por dia"

                menuInicio();

                menuVisualizarAtividadesFisicasDia(diaSelecionado, diasDaSemana, atividadesFisicas, numeroCategorias, atividadesFisicasSelecinadas, numeroAtividadesFisicasSelecionadas);

                // Opcao selecionada pelo usuario

                printf("Entre com a opção: ");

                returnOpcao = scanf("%d", &opcao);

                menuFim();

                // Opções do menu

                opcaoMinMax[0] = 1;

                opcaoMinMax[1] = 1;

                // Limpando o buffer de entrada!!!!

                while(getchar() != '\n');

            // Valida a entrada numerica antes de prosseguir

            } while (!validaEntradaNumerica(opcao, opcaoMinMax, returnOpcao));

            // Verifica a opção escolhida

            if (opcao == 1) {

                    // volta para o menu alterior

                    menu = 1;

                }

        } else if (menu == 2) {

            // Selecionar o dia da semana

            // Menu para a opção 2

            // Passos para chegar a este menu:

            // "Serviços disponíveis:" -> "2 - Registrar uma atividade física"

            // Loop que garante uma entrada numérica válida

            do {

                // Menu "Registrar uma atividade física"

                menuInicio();

                menuRegistrarAtividadeFisica(diasDaSemana);

                // Opcao selecionada pelo usuario

                printf("Entre com a opção: ");

                returnOpcao = scanf("%d", &opcao);

                menuFim();

                // Opções do menu

                opcaoMinMax[0] = 1;

                opcaoMinMax[1] = 8;

                // Limpando o buffer de entrada!!!!

                while(getchar() != '\n');

            // Valida a entrada numerica antes de prosseguir

            } while (!validaEntradaNumerica(opcao, opcaoMinMax, returnOpcao));

            // Verifica qual opção foi escolhida (dia da semana)

            if (opcao >= 1 && opcao <= 7) {

                // Domingo a Sabado - 1 a 7

                diaSelecionado = opcao;

                // Primeiro dígito (2) - "Registrar uma atividade física"

                // Segundo dígito (opcao) - Identifica o dia da semana

                menu = opcao + 20;

                // Atualiza o número de categorias

                for (numeroCategorias = 0; numeroCategorias < MAX\_CATEGORIA; numeroCategorias++) {

                    if(strlen((char\*) atividadesFisicas[numeroCategorias][0]) == 0)

                        break;

                }

            } else {

                // Opcao 8 - Voltar

                // Volta para o menu principal

                menu = 0;

            }

        } else if ( menu >= 21 && menu <= 27) {

            // Selecionar uma categoria ou inserir uma categoria

            // Menu para as opções de 21 a 27

            // Passos para chegar a este menu:

            // "Serviços disponíveis:" -> "2 - Registrar uma atividade física"

            // "Escolha um dia:" -> "1 - Domingo" a "7 - Sábado"

            // Primeiro dígito [2] - Visualizar as atividades físicas

            // Segundo dígito [opcao] - Identifica o diaSelecionado da semana [1-7]

            // Loop que garante uma entrada numérica válida

            do {

                // Menu "Registrar atividades físicas por dia"

                menuInicio();

                menuRegistrarAtividadeFisicaDia(diaSelecionado, diasDaSemana, atividadesFisicas, numeroCategorias);

                printf("Entre com a opção: ");

                // Opcao selecionada pelo usuario

                returnOpcao = scanf("%d", &opcao);

                menuFim();

                // Opções do menu

                opcaoMinMax[0] = 1;

                // Verifica quantas categrias estão cadastradas

                if (numeroCategorias < MAX\_CATEGORIA) {

                    // Número de categorias + opção adicionar categoria + opção voltar

                    opcaoMinMax[1] = numeroCategorias + 2;

                } else {

                    // Número de categorias + opção voltar

                    opcaoMinMax[1] = numeroCategorias + 1;

                }

                // Limpando o buffer de entrada!!!!

                while(getchar() != '\n');

            } while (!validaEntradaNumerica(opcao, opcaoMinMax, returnOpcao));

            // Verifica a opção escolhida

            if (((opcao == numeroCategorias + 1) && (numeroCategorias == MAX\_CATEGORIA)) ||

                ((opcao == numeroCategorias + 2) && (numeroCategorias < MAX\_CATEGORIA))) {

                // Voltar para o menu alterior

                menu = 2;

            // } else if (opcao == numeroCategorias + 1) {

            //     // Adicionar categoria

            } else {

                // Primeiro e segundo dígitos [22] - "Registrar uma atividade física"

                // Terceiro dígito [opcao] - Identifica a categoria

                categoriaSelecionada = opcao;

                menu = categoriaSelecionada + 220;

                // Atualiza o número de atividades físicas

                for (numeroAtividadesFisicas = 0; numeroAtividadesFisicas < MAX\_ATIVIADE\_FISICA; numeroAtividadesFisicas++) {

                    if(strlen((char\*) atividadesFisicas[categoriaSelecionada - 1][numeroAtividadesFisicas + 1]) == 0)

                        break;

                }

            }

        } else if (menu >= 221 && menu <= 220 + numeroCategorias) {

            // Selecionar uma atividade física

            // Menu 221 a [220 + numeroCategorias]

            // Passos para chegar a este menu:

            // "Serviços disponíveis:" -> "2 - Registrar uma atividade física"

            // "Escolha um dia:" -> "1 - Domingo" a "7 - Sábado"

            // "Selecione uma categoria:" -> qualquer categoria disponível

            // Primeiro e segundo dígitos [22] - "Registrar uma atividade física"

            // Terceiro dígito [opcao] - Identifica a categoria

            // Loop que garante uma entrada numérica válida

            do {

                // Menu "Registrar atividades físicas por dia"

                menuInicio();

                menuRegistrarAtividadeFisicaDiaCategoria(diaSelecionado, categoriaSelecionada, diasDaSemana, atividadesFisicas, numeroAtividadesFisicas);

                printf("Entre com a opção: ");

                // Opcao selecionada pelo usuario

                returnOpcao = scanf("%d", &opcao);

                menuFim();

                // Opções do menu

                opcaoMinMax[0] = 1;

                // Verifica quantas atividades físicas estão cadastradas

                if (numeroAtividadesFisicas < MAX\_ATIVIADE\_FISICA) {

                    // Número de atividades físicas + opção adicionar atividade física + opção voltar

                    opcaoMinMax[1] = numeroAtividadesFisicas + 2;

                } else {

                    // Número de categorias + opção voltar

                    opcaoMinMax[1] = numeroAtividadesFisicas + 1;

                }

                // Limpando o buffer de entrada!!!!

                while(getchar() != '\n');

            } while (!validaEntradaNumerica(opcao, opcaoMinMax, returnOpcao));

            // Verifica a opção escolhida

            if (((opcao == numeroAtividadesFisicas + 1) && (numeroAtividadesFisicas == MAX\_ATIVIADE\_FISICA)) ||

                ((opcao == numeroAtividadesFisicas + 2) && (numeroAtividadesFisicas < MAX\_ATIVIADE\_FISICA))) {

                // volta para o menu alterior

                menu = 20 + diaSelecionado;

            //  } else if (opcao == numeroAtividadesFisicas + 1) {

            //      // Adicionar uma nova atividade física

            } else {

                // Primeiro, segundo e terceiro dígito (222) - "Registrar uma atividade física"

                // Quarto dígito (opcao) - Identifica a categoria

                atividadeSelecionada = opcao;

                menu = atividadeSelecionada + 2220;

            }

        } else if (menu == 220 + numeroCategorias + 1) {

            // Adicionar uma nova categoria

            // Menu [220 + numeroCategorias + 1]

            // "Serviços disponíveis:" -> "2 - Registrar uma atividade física"

            // "Escolha um dia:" -> "1 - Domingo" a "7 - Sábado"

            // "Selecione uma categoria:" -> "Adicionar uma categoria"

            // Primeiro e segundo dígitos [22] - Visualizar as atividades físicas

            // Terceiro dígito [numeroCategorias + 1] - Adicionar uma nova categoria

            // Loop que garante uma entrada numérica válida

            do {

                // Menu "Registrar atividades físicas por dia"

                // Adicionar uma nova categoria

                menuInicio();

                menuRegistrarAtividadeFisicaDiaNovaCategoria(diaSelecionado, diasDaSemana, atividadesFisicas, numeroCategorias);

                printf("Digite uma nova categoria: ");

                // Nova categoria digitada pelo usuário

                // A função fgets foi utilizada para obter strings com espaços.

                returnNovaCategoria = fgets(novaCategoria, 50, stdin);

                menuFim();

                // Eliminando '\n' da string

                novaCategoria[strcspn(novaCategoria, "\n")] = 0;

            } while (!validaEntradaCaracters(novaCategoria, returnNovaCategoria));

            if (strlen(novaCategoria) > 0) {

                // Adicina a nova categoria ao array de atividadesFisicas

                //numeroCategorias = categoria;

                numeroCategorias++;

                strcpy(atividadesFisicas[numeroCategorias-1][0], novaCategoria);

                // Imprime uma mensagem confirmando os dados adicionados

                printf("\n -> Nva categoria adicionada!\n");

                printf(" [Nova categoria: %s]\n", atividadesFisicas[numeroCategorias-1][0]);

            }

            // Retorna ao menu anterior

             menu = 20 + diaSelecionado;

        } else if (menu >= 2221 && menu <= 2220 + numeroAtividadesFisicas) {

            // Selecionar uma atividade física

            // Menu para as opções 2221 a [2220 + numeroAtividadesFisicas]

            // Passos para chegar a este menu:

            // "Serviços disponíveis:" -> "2 - Registrar uma atividade física"

            // "Escolha um dia:" -> "1 - Domingo" a "7 - Sábado"

            // "Selecione uma categoria:" -> qualquer categoria disponível

            // "Selecione uma atividade física:" -> qualquer atividade física disponivel

            // Verifica se o número máximo de registro de atividades físicas não foi atingido

            if (numeroAtividadesFisicasSelecionadas < MAX\_REGISTRO\_ATIVIDADE) {

                // Atualiza o número de atividades físicas selecinadas

                numeroAtividadesFisicasSelecionadas++;

                // Inclui a atividade física selecinada ao vetor de atividades físicas

                atividadesFisicasSelecinadas[numeroAtividadesFisicasSelecionadas - 1][0] = diaSelecionado - 1;

                atividadesFisicasSelecinadas[numeroAtividadesFisicasSelecionadas - 1][1] = categoriaSelecionada - 1;

                atividadesFisicasSelecinadas[numeroAtividadesFisicasSelecionadas - 1][2] = atividadeSelecionada;

                // Imprime uma mensagem confirmando os dados adicionados

                printf("\n -> Atividade física adicionada!\n");

                printf(" [Dia: %s][Categoria: %s][Atividade física: %s]\n",

                    diasDaSemana[diaSelecionado - 1],

                    atividadesFisicas[categoriaSelecionada - 1],

                    atividadesFisicas[categoriaSelecionada - 1][atividadeSelecionada]

                );

            }

                // Voltar para o menu principal

            menu = 0;

        } else if (menu == 2220 + numeroAtividadesFisicas + 1) {

            // Adicinar uma atividade física

            // Menu para a opção [2220 + numeroAtividadeFisica + 1]

            // Passos para chegar a este menu:

            // "Serviços disponíveis:" -> "2 - Registrar uma atividade física"

            // "Escolha um dia:" -> "1 - Domingo" a "7 - Sábado"

            // "Selecione uma categoria:" -> qualquer categoria disponível

            // "Selecione uma atividade física:" -> Adicionar uma atividade física

            // Primeiro, segundo e terceiro dígitos [222] - Visualizar as atividades físicas

            // Quarto dígito [numeroAtividadesFisicas + 1] - Adicionar uma nova atividade física

            // Loop que garante uma entrada de caracteres válida

            do {

                // Adicionar uma nova atividade física

                menuInicio();

                menuRegistrarAtividadeFisicaDiaNovaAtividadeFisica(diaSelecionado, categoriaSelecionada, diasDaSemana, atividadesFisicas, numeroCategorias);

                printf("Digite uma nova atividade física: ");

                // Nova atividade física digitada pelo usuário

                // A função fgets foi utilizada para obter strings com espaços.

                returnNovaAtividadeFisica = fgets(novaAtividadeFisica, 50, stdin);

                menuFim();

                // fgets retrna uma string com '\n' no final

                // Retirando '\n' do final da string

                novaAtividadeFisica[strcspn(novaAtividadeFisica, "\n")] = 0;

            } while (!validaEntradaCaracters(novaAtividadeFisica, returnNovaAtividadeFisica));

            if (strlen(novaAtividadeFisica) > 0) {

                // Adiciona a nova atividade fisica ao array de atividadesFisicas

                numeroAtividadesFisicas++;

                strcpy(atividadesFisicas[categoriaSelecionada-1][numeroAtividadesFisicas], novaAtividadeFisica);

                // Imprime uma mensagem confirmando os dados adicionados

                printf("\n -> Nova atividade física adicionada!\n");

                printf(" [Nova atividade física: %s]\n", atividadesFisicas[categoriaSelecionada-1][numeroAtividadesFisicas]);

            }

            // Retorna ao menu anterior

             menu = 220 + categoriaSelecionada;

        } else {

            // Menu principal

            menu = 0;

        }

        // Sair - sairMenu igual a 1 encerra o sistema!

    } while (!sairMenu);

    mensagemEncerramento();

    return 0;

}

void apresentacaoSistema() {

    // Apresnetação do sistema

    printf("\n");

    printf("####################################################################\n");

    printf("########## Sistema de Registro de Atividades Físicas v1.0 ##########\n");

    printf("####################################################################\n");

    printf("Sobre:\n");

    printf("Desenvolvido por Fabio Toledo Bonemer De Salvi\n");

    printf("\n");

    printf("\* Visualize as suas atividades físicas diárias!\n");

    printf("-> Visualize as atividades físicas por dia da semana ou pela semana\n");

    printf("    inteira!\n");

    printf("\n");

    printf("\* Registre as suas atividades físicas diárias\n");

    printf("-> Escolha um dia da semana, uma categoria e uma atividade física!\n");

    printf("-> Adicione novas categorias!\n");

    printf("-> Adicione novas atividades físicas!\n");

    printf("####################################################################\n");

}

void menuInicio() {

    printf("\n");

    printf("####################################################################\n");

    printf("########## Sistema de Registro de Atividades Físicas v1.0 ##########\n");

    printf("####################################################################\n");

}

void menuFim() {

    printf("####################################################################\n");

}

void menuPrincipal() {

    // Menu 0 - Menu principal

    // Menu inicial

    printf("\n");

    printf(">> Serviços disponíveis:\n");

    printf("1 - Visualizar as atividades físicas\n");

    printf("2 - Registrar uma atividade física\n");

    printf("3 - Sair\n");

}

void menuVisualizarAtividadesFisicas(char diasDaSemana[7][15]) {

    // Menu 1 - Opção 1 do menu principal

    // Vizualizar as atividades físicas

    printf("================= Visualizar as Atividades Físicas =================\n\n");

    printf(">> Visualizar as atividades físicas de qual(quais) dia(s)?\n");

    // Imprime os dias da semana

    for (int indiceDia = 0; indiceDia < strlen((char\*)diasDaSemana); indiceDia++)

        printf("%d - %s\n", indiceDia + 1, diasDaSemana[indiceDia]);

    printf("8 - Semana inteira\n");

    printf("9 - Voltar\n");

}

void menuVisualizarAtividadesFisicasDia(int diaSelecionado, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroCategorias, int atividadesFisicasSelecinadas[50][3], int numeroAtividadesFisicasSelecionadas) {

    // Menu  11 ao 18 - Vizualizar as atividades físicas por dia

    // Opção 1 a 7 - [Domingo ao Sábado]

    // ou

    // Opção 8 - [Semana inteira]

    printf("================= Visualizar as Atividades Físicas =================\n\n");

    // Número de atividades por dia -> posições de 0 a 6

    // Maior atividade -> posição 7

    int numeroAtividadesDia[8] = {0, 0, 0, 0, 0, 0, 0, 0};

    for (int indexDia = 0; indexDia <= 6; indexDia++) {

        int indexAtividade;

        for(indexAtividade = 0; indexAtividade < numeroAtividadesFisicasSelecionadas; indexAtividade++) {

                if (indexDia == atividadesFisicasSelecinadas[indexAtividade][0])

                    numeroAtividadesDia[indexDia]++;

        }

    }

    // Determinando o(s) Top Day

    for (int indexDia = 0; indexDia <= 6; indexDia++) {

        for (int indexNumeroAtividades = indexDia; indexNumeroAtividades < 7; indexNumeroAtividades++) {

            if (numeroAtividadesDia[indexDia] > numeroAtividadesDia[7])

                numeroAtividadesDia[7] = numeroAtividadesDia[indexDia];

        }

    }

    // Verificando se o valor Top Day é único

    int topDayUnico = 0;

    for (int indexDia = 0; indexDia <= 6; indexDia++) {

        if (numeroAtividadesDia[indexDia] == numeroAtividadesDia[7])

            topDayUnico++;

        // Se o TopDay não for único a maior atividade é retirada do vetor numero de atividades

        if (topDayUnico != 1 && indexDia == 6)

            numeroAtividadesDia[7] = 0;

    }

    // Imprime as atividades físicas para a semana inteira

    for (int indexDia = 0; indexDia <= 6; indexDia++) {

        if ((diaSelecionado == 8) || (indexDia == diaSelecionado - 1)) {

            if (indexDia > 0 && indexDia < 6)

                printf(">> Atividades físicas da %s:\n", diasDaSemana[indexDia]);

            else

                printf(">> Atividades físicas do %s:\n", diasDaSemana[indexDia]);

            // Verifica se é o Top Day!

            if (numeroAtividadesDia[indexDia] == numeroAtividadesDia[7] && numeroAtividadesDia[7]) {

                printf(" > TOP DAY!!\n");

            }

            int indexAtividade;

            for(indexAtividade = 0; indexAtividade < numeroAtividadesFisicasSelecionadas; indexAtividade++) {

                if (indexDia == atividadesFisicasSelecinadas[indexAtividade][0]) {

                    // Imprime a

                    printf(" - %s (%s)\n",

                        atividadesFisicas

                            [atividadesFisicasSelecinadas[indexAtividade][1]]

                            [atividadesFisicasSelecinadas[indexAtividade][2]],

                        atividadesFisicas

                            [atividadesFisicasSelecinadas[indexAtividade][1]]

                    );

                }

                // // Imprime o número de atividades ao chegar no final do vetor

                // if (indexAtividade == numeroAtividadesFisicasSelecionadas - 1) {

                //     printf(" \* %d atividade(s)!\n", numeroAtividadesDia[indexDia]);

                // }

            }

            // Imprime o número de atividades ao chegar no final do vetor

            printf(" \* %d atividade(s)!\n\n", numeroAtividadesDia[indexDia]);

        }

    }

    // Imprime as atividades físicas

    printf("1 - Voltar\n");

}

void menuRegistrarAtividadeFisica(char diasDaSemana[7][15]) {

    // Menu 2 - Opção 2 do menu menu principal

    // Registrar um atividade física

    // Mostra os dias da semana

    printf("================== Registrar uma atividade física ==================\n\n");

    printf(">> Escolha um dia:\n");

    // Imprime os dias da semana

    for (int indiceDia = 0; indiceDia < strlen((char\*)diasDaSemana); indiceDia++)

        printf("%d - %s\n", indiceDia + 1, diasDaSemana[indiceDia]);

    printf("8 - Voltar\n");

}

void menuRegistrarAtividadeFisicaDia(int diaSelecionado, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroCategorias) {

    // Registrar uma atividade física (Menu 21 a 27)

    // Mostra o dia selecionado

    // Mostra as categorias cadastradas

    // Mostra a opção de adicionar categoria

    printf("================== Registrar uma atividade física ==================\n\n");

    // Imprime o dia da semana selecionado

    printf(">> Dia selecionado: %s.\n\n", diasDaSemana[diaSelecionado-1]);

    printf(">> Selecione uma categoria:\n");

    // Imprime as categorias de atividade física

    for (int indiceCategoria = 0;

            indiceCategoria < numeroCategorias;

            indiceCategoria++)

        printf("%d - %s\n", indiceCategoria + 1, atividadesFisicas[indiceCategoria][0]);

    // Imprime a opção de adicionar categoria

    if (numeroCategorias < MAX\_CATEGORIA)

        printf("%d - Adicionar uma categoria\n", numeroCategorias + 1 );

    // Imprime a opção voltar

    printf("%d - Voltar\n", (numeroCategorias < MAX\_CATEGORIA ) ? numeroCategorias + 2 : numeroCategorias + 1);

}

void menuRegistrarAtividadeFisicaDiaCategoria(int diaSelecionado, int categoriaSelecionada, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroAtividadesFisicas) {

    // Registrar uma atividade física (menu = 2)

    // Opção 1 - Segunda-Feira

    printf("================== Registrar uma atividade física ==================\n\n");

    // Imprime o dia da semana selecionado

    printf(">> Dia selecionado: %s.\n", diasDaSemana[diaSelecionado-1]);

    // Imprime a categoria selecionada

    printf(">> Categoria selecionada: %s.\n\n", atividadesFisicas[categoriaSelecionada - 1][0]);

    printf(">> Selecione uma atividade física:\n");

    // Imprime as atividades físicas

    for (int indiceAtividadeFisica = 1;

            indiceAtividadeFisica <= numeroAtividadesFisicas;

            indiceAtividadeFisica++)

        printf("%d - %s\n", indiceAtividadeFisica, atividadesFisicas[categoriaSelecionada - 1][indiceAtividadeFisica]);

    // Imprime a opção de adicionar atividade física

    if (numeroAtividadesFisicas < MAX\_ATIVIADE\_FISICA)

        printf("%d - Adicionar uma atividade física\n", numeroAtividadesFisicas + 1);

    printf("%d - Voltar\n", (numeroAtividadesFisicas < MAX\_ATIVIADE\_FISICA)? numeroAtividadesFisicas + 2 : numeroAtividadesFisicas + 1);

}

void menuRegistrarAtividadeFisicaDiaNovaCategoria(int diaSelecionado, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroAtividadesFisicas) {

    // Registrar uma atividade física (menu = 2)

    printf("================== Registrar uma atividade física ==================\n\n");

    // Imprime o dia da semana selecionado

    printf(">> Dia selecionado: %s.\n\n", diasDaSemana[diaSelecionado - 1]);

    // Imprime a mensagem nova categoria

    printf(">> Nova categoria.\n");

}

void menuRegistrarAtividadeFisicaDiaNovaAtividadeFisica(int diaSelecionado, int categoriaSelecionada, char diasDaSemana[7][15], char atividadesFisicas[10][11][50], int numeroAtividadesFisicas) {

    // Registrar uma atividade física (menu = 2)

    // Opção 1 - Segunda-Feira

    printf("================== Registrar uma atividade física ==================\n\n");

    // Imprime o dia da semana selecionado

    printf(">> Dia selecionado: %s.\n", diasDaSemana[diaSelecionado-1]);

    // Imprime a categoria selecionada

    printf(">> Categoria selecionada: %s.\n\n", atividadesFisicas[categoriaSelecionada - 1][0]);

    // Imprime a mensagem nova atividade física

    printf(">> Nova atividade física.\n");

}

int validaEntradaNumerica(int opcao, int opcaoMinMax[2], int returnOpcao) {

    // Verifica se apenas um valor inteiro foi digitado

    if (returnOpcao != 1) {

        printf("\nValor inválido!\n");

        return 0; // Entrada inválida

    }

    // Verifica se a opção digitada esta disponível no menu

    if (opcao < opcaoMinMax[0] || opcao > opcaoMinMax[1]) {

        printf("\nOpção inválida!\n");

        return 0; // Opção inválida

    }

    return 1; // Entrada numérica Ok

}

int validaEntradaCaracters(char \*caracteres, char \*returnNovoCaracteres) {

    // Verifica se apenas um valor de caracteres foi digitado

    if (!(strlen(returnNovoCaracteres) > 0)) {

        printf("\nValor inválido!\n");

        return 0; // Entrada inválida

    }

    return 1; // Entrada caracteres Ok

}

void mensagemEncerramento() {

    // Mensagem de encerramento

    printf("\n");

    printf("####################################################################\n");

    printf("########## Sistema de Registro de Atividades Físicas v1.0 ##########\n");

    printf("####################################################################\n");

    printf("Obrigado por utilizar nosso sistema!\n\n");

    printf("Volte sempre!\n");

    printf("####################################################################\n");

}

// Remove espaços em branco do inicio da string

void trim(char \*s) {

    int i = 0;

    // Find the first non-whitespace character

    while (isspace((unsigned char)s[i])) {

        i++;

    }

    // Shift the remaining characters to the beginning of the string

    if (i > 0) {

        int j = 0;

        while ((s[j++] = s[i++]) != '\0');

    }

}

// run into prompt:

// $ prompt $G

// $ exemplo{number}.exe

// $ echo %errorlevel%

// Como verificar o tamanho de arrays

// printf( %d - %d - %d - %d - %d", numeroCategorias, sizeof(atividadesFisicas[0]), sizeof(atividadesFisicas[0][0]), sizeof(atividadesFisicas[0][0][0]), sizeof(atividadesFisicas[0][0][0][0]));

# Teste de Execução

O software desenvolvido foi compilado e executado em prompt de comandos utilizando o software *VS Code* ([MICROSOFT CORPORATION, 2025](https://code.visualstudio.com/)).

A execução do software com as devidas entradas de dados e mensagens mostradas ao usuário serão dispostas nos tópicos subsequentes.

As opções foram escolhidas digitando os números nas respectivas entradas numéricas.

Para as novas categorias e novas atividades físicas foram digitados os textos nas respectivas entradas de caracteres.

## Apresentação do sistema

Ao iniciar a execução do software a seguinte mensagem de apresentação é apresentada no prompt de comandos:

Tabela

O conteúdo gerado por IA pode estar incorreto.

Figura - Tela de apresentação do software.

## Menu dos serviços disponíveis

O menu de opções é apresentado na após a apresentação do sistema e mostra as opções de visualizar as atividades físicas, de registrar uma atividade física e de sair do programa.

Para continuar é preciso digitar o valor numérico referente à opção desejada.

Texto

O conteúdo gerado por IA pode estar incorreto.

Figura - Menu dos serviços disponíveis

## Visualização das atividades físicas antes de incluir atividades físicas

A visualização das atividades físicas no início resultara em valores nulos para todos os dias da semana.

Foi escolhido a opção “**1 – Visualizar as atividades físicas**” no menu de “**Serviços disponíveis**” para a visualização das atividades físicas.

No menu “**Visualizar as atividades físicas de qual(quais) dia(s)?**” o usuário deve escolher entre **um dia da semana** ou **semana inteira**.

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

Figura - Menu de escolha do dia ou da semana inteira

A escolha da opção “**1 – Domingo**” resulta em uma mensagem com valor nulo para o número de atividades.

Texto

O conteúdo gerado por IA pode estar incorreto.

Figura - Atividades físicas realizadas no domingo.

E a escolha da opção “**8 – Semana inteira**” resulta em uma mensagem com valor nulo para o número de atividades para cada dia da semana.

Tabela

O conteúdo gerado por IA pode estar incorreto.

Figura - Atividades físicas realizadas na semana inteira.

O retorno ao menu principal contendo os “**Serviços disponíveis**” foi realizada digitando os valores correspondentes às opções “**1 – Voltar**” e “**9 – Voltar”**.

## Registrar uma atividade física

A inclusão das atividades físicas foi realizada selecionando a opção “**2 – Registrar uma atividade física**” no menu principal dos “**Serviços disponíveis”**.

No menu “**Escolha um dia:**” foi escolhido o dia “**4 – Quarta-Feira**”.

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Figura - Menu para escolha do dia da semana que será inserido a atividade física.

Texto

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

Uma imagem contendo Gráfico

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Uma imagem contendo Logotipo

O conteúdo gerado por IA pode estar incorreto.







|  |  |  |
| --- | --- | --- |
| Dia da Semana | Categoria | Exercício |
| 4 - Quarta-Feira | 3 - Exercícios de Flexibilidade / Alongamento | 2 - Alongamento de braços e ombros |
| 6 - Sexta-Feira | 2 - Força / Musculação | 7 - Prancha (Plank) - para core |
| 3 - Terça-Feira | 5 - Atividades de Equilíbrio / Core | 3 - Abdominais (Crunches, Bicycle crunch) |
| 7 - Sábado | 4 - Exercícios Funcionais / HIIT | 3 - Jump squats |
| 1 - Domingo | 1 - Cardiorespiratórios | 5 - Natação |
| 5 - Quinta-Feira | 2 - Força / Musculação | 6 - Remada (Row) |

## Adicionando uma categoria e uma atividade física

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.



Texto

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.



Uma imagem contendo Texto

O conteúdo gerado por IA pode estar incorreto.





Uma imagem contendo Logotipo

O conteúdo gerado por IA pode estar incorreto.

|  |  |  |
| --- | --- | --- |
| Dia da Semana | Categoria | Exercício |
| 2 - Segunda-Feira | 6 - Atividades Esportivas Recreativas | 1 – Basquete |
| 4 - Quarta-Feira | 7 - Exercícios de Mobilidade Articular | 1 - Rotação de ombros |

## Adicionando uma nova atividade física

Texto

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.

## Mensagem de encerramento do programa

Texto

O conteúdo gerado por IA pode estar incorreto.

# Conclusão

O objetivo desta experiência prática ao desenvolver um fluxograma que computasse e analisasse a variação de preços dos produtos de uma cesta básica era apresentar um caso real para aplicação de um algoritmo computacional.

Com o sucesso dos resultados do computo e análise dos dados de variação de preços é possível afirmar que o objetivo de apresentar o pensamento computacional e exercitar a lógica de programação teve êxito.

# Referências

MICROSOFT. *Using GCC with MinGW*. Disponível em: [https://code.visualstudio.com/docs/cpp/config-mingw](https://code.visualstudio.com/docs/cpp/config-mingw?utm_source=chatgpt.com). Acesso em: 28 set. 2025.

MICROSOFT CORPORATION. *Visual Studio Code*. Versão 1.93. Redmond: Microsoft, 2025. Disponível em: <https://code.visualstudio.com/>. Acesso em: 28 set. 2025.

MINGW-W64 PROJECT. *MinGW-w64: Minimalist GNU for Windows*. [S.l.]: MinGW-w64, 2025. Disponível em: <https://www.mingw-w64.org/>. Acesso em: 28 set. 2025.

UNIVERSIDADE DE FRANCA (UNIFRAN). *Experiências Práticas – Fundamentos da Introdução à Programação e ao Pensamento Computacional*. Franca: UNIFRAN, [2025]. Apostila da disciplina Algoritmos e Pensamento Computacional, Curso EAD. Disponível em: <https://www.unifran.edu.br/>. Acesso em: 28 set. 2025.

SALVI, Fabio Toledo Bonemer De. *basic-food-basket-flowchart*. GitHub, 2025. Disponível em: <https://github.com/engfabiodesalvi/physical-activity-recording-system.git>. Acesso em: 28 set. 2025.