

```

1: //Disciplina: Programação Estruturada e Modular
2: // Prof. Carlos Veríssimo.
3: //Objetivo: Programa para cálculo de notas
4: //Data: 23/08/2024
5: //Autor: Cyntia Farias Ruffo
6:
7: #include <stdio.h>
8: #define MAX_NOTAS 100
9: #define MAX_TESTES 100
10: int main() {
11:     int N;
12:     float notas[MAX_NOTAS];
13:     float nota;
14:     int i;
15:     float min, max;
16:     int min_index, max_index;
17:     int continuar;
18:     float total_teste;
19:     float totais_testes[MAX_TESTES]; // Array para armazenar os totais dos testes
20:     int num_testes = 0;
21:     float total_geral = 0;
22:     do {
23:         // Inicializa variáveis para o novo teste
24:         total_teste = 0;
25:         min_index = max_index = -1;
26:         // Solicita o valor de N e garante que seja maior que 2
27:         do {
28:             printf("\nInsira o valor de N: (OBS: N > 2) ");
29:             scanf("%d", &N);
30:             if (N <= 2) {
31:                 printf("INSIRA NOVAMENTE! O VALOR DE N DEVE SER > 2!\n");
32:             }
33:         } while (N <= 2);
34:         // Coleta as notas
35:         for (i = 0; i < N; i++) {
36:             while (1) { // Loop para garantir que a nota esteja entre 0 e 10
37:                 printf("Insira a %d nota: ", i + 1);
38:                 scanf("%f", &nota);
39:                 if (nota >= 0 && nota <= 10) {
40:                     break; // Nota válida, sai do loop
41:                 } else {
42:                     printf("NOTA INVÁLIDA! INSIRA NOVAMENTE!\n");
43:                 }
44:             }
45:             notas[i] = nota;
46:         }
47:
48:         // Imprime as notas
49:         printf("\nNotas inseridas:\n");
50:         for (i = 0; i < N; i++) {
51:             printf("%.2f ", notas[i]);
52:         }
53:         printf("\n");
54:         // Inicializa min e max com o primeiro valor do array
55:         if (N > 0) {
56:             min = max = notas[0];
57:             min_index = max_index = 0;
58:             // Encontra a menor e a maior nota
59:             for (i = 1; i < N; i++) {
60:                 if (notas[i] < min) {

```

```

61: min = notas[i];
62: min_index = i;
63: }
64: if (notas[i] > max) {
65: max = notas[i];
66: max_index = i;
67: }
68: }
69: // Verifica se a menor e maior nota foram encontrados
70: if (min_index == -1 || max_index == -1) {
71: printf("Erro: Não foi possível encontrar a menor ou a maior nota.\n");
72: return 1; // Código de erro
73: }
74: // Cria um novo array para armazenar as notas restantes
75: float notas_restantes[MAX_NOTAS];
76: int j = 0;
77: // Copia as notas, excluindo a menor e a maior
78: for (i = 0; i < N; i++) {
79: if (i != min_index && i != max_index) {
80: notas_restantes[j++] = notas[i];
81: }
82: }
83: // Atualiza o número de notas restantes
84: int N_restantes = j;
85:
86: // Imprime as notas restantes
87: printf("\nNotas restantes (sem a menor e a maior nota):\n");
88: for (i = 0; i < N_restantes; i++) {
89: printf("%.2f ", notas_restantes[i]);
90: }
91: printf("\n");
92:
93: // Calcula o total do teste
94: for (i = 0; i < N_restantes; i++) {
95: total_teste += notas_restantes[i];
96: }
97:
98: printf("\nTotal do Teste: %.2f", total_teste);
99:
100: // Atualiza o total geral e armazena o total do teste
101: total_geral += total_teste;
102: totais_testes[num_testes] = total_teste;
103: num_testes++;
104:
105: // Pergunta ao usuário se deseja continuar
106: printf("\n\nDeseja realizar outro teste? (1 - Sim, 0 - Não): ");
107: scanf("%d", &continuar);
108: } else {
109: printf("Erro: O array de notas está vazio.\n");
110: return 1; // Código de erro
111: }
112: } while (continuar);
113:
114: // Exibe os totais finais quando o usuário decide parar
115: for (i = 0; i < num_testes; i++) {
116: printf("Final do teste %d: %.2f\n", i + 1, totais_testes[i]);
117: }
118: printf("Total Geral: %.2f\n", total_geral);
119: printf("Número de Testes: %d\n", num_testes);
120: return 0;

```

121: }