

```

1: #include <stdio.h>
2: #include <stdlib.h>
3: #include <math.h>
4:
5: /**
6:  * @author Raique Carvalho Queiroz - Eng Computação - 162030146
7:  * @content Lista Sequencial
8:  * @language C
9:  * @date 2019 February
10: */
11:
12: /* //NOTA//
13:  SE O PROJETO FOR RODAR NO ONLINEGBD AS FUNCOES "SYSTEM()" NAO FUNCIONARAO!
14: */
15:
16: #define MAX 5
17:
18: typedef int TIPOCHAVE;
19:
20: typedef struct{
21:     TIPOCHAVE chave;
22: } REGISTRO;
23:
24: typedef struct{
25:     REGISTRO vetor[MAX];
26:     int nroElem;
27: } LISTA;
28:
29: void criarLista(LISTA *lista);
30: void reiniciarLista(LISTA *lista);
31:
32: void inserirFinal(LISTA *lista, REGISTRO registro);
33: void inserirPosicao(LISTA *lista, REGISTRO registro, int posicao);
34: void inserirInicio(LISTA *lista, REGISTRO registro);
35:
36: void imprimirTamanho(LISTA lista);
37: void imprimirPosicao(LISTA *lista, int posicao);
38: void imprimirElemento(LISTA *lista, TIPOCHAVE chave);
39: void imprimirListaInicioFim(LISTA *lista);
40: void imprimirListaFimInicio(LISTA *lista);
41:
42: void deletarPosicao(LISTA *lista, int posicao);
43:
44: int main(){
45:
46:     LISTA lista;
47:     REGISTRO registro;
48:     TIPOCHAVE chave;
49:     int opc, posicao, opcAux;
50:     opc=0;
51:
52:     do{
53:         system("cls");
54:         printf("Seja bem vindo ao Menu!\n");
55:         printf("Selecione uma das opcoes abaixo:\n\n");
56:         printf("1 - Criar Estrutura\n");
57:         printf("9 - Sair\n");
58:         printf("Opcao: ");
59:         scanf("%d",&opc);
60:         switch(opc){

```

```

61:         case 1:{
62:             criarLista(&lista);
63:             break;
64:         }
65:         case 9:{
66:             exit(1);
67:             break;
68:         }
69:         default:{
70:             system("cls");
71:             printf("Opcao Invalida!\nVoce Voltara ao Menu!\n");
72:             system("pause");
73:             break;
74:         }
75:     }
76: }while(opc!=1);
77:
78: do{
79:     system("cls");
80:     printf("Seja bem vindo ao Menu!\n");
81:     printf("Selecione uma das opcoes abaixo:\n\n");
82:     printf("1 - Quantidade de Registros Validos\n");
83:     printf("2 - Exibir Registros\n");
84:     printf("3 - Buscar\n");
85:     printf("4 - Inserir Registro\n");
86:     printf("5 - Excluir Registro\n");
87:     printf("6 - Reinicializar a Estrutura\n");
88:     printf("9 - Sair\n");
89:     printf("Opcao: ");
90:     scanf("%d",&opc);
91:     switch(opc){
92:         case 1:{
93:             imprimirTamanho(lista);
94:             break;
95:         }
96:         case 2:{
97:             system("cls");
98:             printf("Selecione uma das opcoes abaixo: \n\n");
99:             printf("1 - Exibir do Inicio ao Fim\n");
100:            printf("2 - Exibir do Fim ao Inicio\n");
101:            printf("Opcao: ");
102:            scanf("%d",&opcAux);
103:            system("cls");
104:            switch(opcAux){
105:                case 1:{
106:                    imprimirListaInicioFim(&lista);
107:                    break;
108:                }
109:                case 2:{
110:                    imprimirListaFimInicio(&lista);
111:                    break;
112:                }
113:                default:{
114:                    printf("Opcao Invalida!\nVoce Voltara ao Menu!\n");
115:                    system("pause");
116:                    break;
117:                }
118:            }
119:            break;
120:        }

```

```

121:         case 3:{
122:             system("cls");
123:             printf("Selecione uma das opcoes abaixo: \n\n");
124:             printf("1 - Buscar por Chave\n");
125:             printf("2 - Buscar por Posicao\n");
126:             printf("Opcao: ");
127:             scanf("%d",&opcAux);
128:             system("cls");
129:             switch(opcAux){
130:                 case 1:{
131:                     printf("Informe a Chave do Registro que deseja Buscar: ");
132:                     scanf("%d",&chave);
133:                     imprimirElemento(&lista,chave);
134:                     break;
135:                 }
136:                 case 2:{
137:                     printf("Informe a Posicao do Registro que deseja Buscar: ");
138:                     scanf("%d",&posicao);
139:                     imprimirPosicao(&lista,posicao);
140:                     break;
141:                 }
142:                 default:{
143:                     printf("Opcao Invalida!\nVoce Voltara ao Menu!\n");
144:                     system("pause");
145:                     break;
146:                 }
147:             }
148:             break;
149:         }
150:         case 4:{
151:             system("cls");
152:             printf("Selecione uma das opcoes abaixo: \n\n");
153:             printf("1 - Inserir no Inicio\n");
154:             printf("2 - Inserir no Fim\n");
155:             printf("3 - Inserir por Posicao\n");
156:             printf("Opcao: ");
157:             scanf("%d",&opcAux);
158:             system("cls");
159:             switch(opcAux){
160:                 case 1:{
161:                     printf("Informe a Chave do Registro: ");
162:                     scanf("%d",&registro.chave);
163:                     inserirInicio(&lista,registro);
164:                     break;
165:                 }
166:                 case 2:{
167:                     printf("Informe a Chave do Registro: ");
168:                     scanf("%d",&registro.chave);
169:                     inserirFinal(&lista,registro);
170:                     break;
171:                 }
172:                 case 3:{
173:                     printf("Informe a Chave do Registro: ");
174:                     scanf("%d",&registro.chave);
175:                     printf("Informe a Posicao: ");
176:                     scanf("%d",&posicao);
177:                     inserirPosicao(&lista,registro,posicao);
178:                     break;
179:                 }
180:                 default:{

```

```

181:             printf("Opcao Invalida!\nVoce Voltara ao Menu!\n");
182:             system("pause");
183:             break;
184:         }
185:     }
186:     break;
187: }
188: case 5:{
189:     system("cls");
190:     printf("Informe a Posicao do Registro: ");
191:     scanf("%d",&posicao);
192:     deletarPosicao(&lista,posicao);
193:     break;
194: }
195: case 6:{
196:     reiniciarLista(&lista);
197:     break;
198: }
199: case 9:{
200:     exit(1);
201:     break;
202: }
203: default:{
204:     system("cls");
205:     printf("Opcao Invalida!\nVoce Voltara ao Menu!\n");
206:     system("pause");
207:     break;
208: }
209: }
210: }while(opc!=9);
211: return 0;
212: }
213:
214: void criarLista(LISTA *lista){
215:     lista->nroElem = 0;
216:     system("cls");
217:     printf("Sucesso!\n");
218:     system("pause");
219: }
220:
221: void reiniciarLista(LISTA *lista){
222:     int opc;
223:     system("cls");
224:     printf("Voce tem certeza que deseja Reiniciar a Estrutura?\n");
225:     printf("0 - SIM\n");
226:     printf("1 - NAO\n");
227:     printf("Opcao: ");
228:     scanf("%d",&opc);
229:     system("cls");
230:     switch(opc){
231:         case 0:{
232:             lista->nroElem = 0;
233:             printf("Sucesso!\n");
234:             break;
235:         }case 1:{
236:             printf("Voce voltara ao Menu!\n");
237:             break;
238:         }
239:         default:{
240:             printf("Opcao Invalida!\nVoce Voltara ao Menu!\n");

```

```

241:         break;
242:     }
243: }
244:
245:     system("pause");
246: }
247:
248: int verificaChave(LISTA *lista, REGISTRO registro){
249:     int i;
250:     for(i=0;i<lista->nroElem;i++)
251:         if(registro.chave==lista->vetor[i].chave)
252:             return 1;
253:     return 0;
254: }
255:
256: void inserirFinal(LISTA *lista, REGISTRO registro){
257:     int verificador = verificaChave(lista,registro);
258:     system("cls");
259:     if(verificador==0){
260:         if(lista->nroElem<MAX){
261:             lista->vetor[lista->nroElem] = registro;
262:             lista->nroElem = lista->nroElem+1;
263:             printf("Sucesso!\n");
264:         }else{
265:             printf("Erro!\n");
266:             printf("Lista Cheia!\n");
267:         }
268:     }else{
269:         printf("Erro!\n");
270:         printf("Chave de Registro ja inserida na lista!\n");
271:     }
272:     system("pause");
273: }
274:
275: void inserirPosicao(LISTA *lista, REGISTRO registro, int posicao){
276:     int i;
277:     int verificador = verificaChave(lista,registro);
278:     if(verificador==0){
279:         if(posicao>=lista->nroElem)
280:             inserirFinal(lista,registro);
281:         else{
282:             system("cls");
283:             if(lista->nroElem<MAX){
284:                 printf("Sucesso!\n");
285:                 for(i=lista->nroElem;i>posicao;i--)
286:                     lista->vetor[i] = lista->vetor[i-1];
287:                 lista->vetor[posicao] = registro;
288:                 lista->nroElem++;
289:             }else{
290:                 printf("Erro!\n");
291:                 printf("Lista Cheia!\n");
292:             }
293:             system("pause");
294:         }
295:     }else{
296:         system("cls");
297:         printf("Erro!\n");
298:         printf("Chave de Registro ja inserida na lista!\n");
299:         system("pause");
300:     }

```

```

301: }
302:
303: void inserirInicio(LISTA *lista, REGISTRO registro){
304:     inserirPosicao(lista,registro,0);
305: }
306:
307: void imprimirTamanho(LISTA lista){
308:     system("cls");
309:     if(lista.nroElem==0)
310:         printf("Lista Vazia!\n");
311:     else
312:         printf("Quantidade: %d\n",lista.nroElem);
313:     system("pause");
314: }
315:
316: void imprimirPosicao(LISTA *lista, int posicao){
317:     system("cls");
318:     if(posicao<lista->nroElem)
319:         printf("O Registro da posicao %d: %d\n",posicao,lista->vetor[posicao].chave);
320:     else
321:         printf("Posicao nao existe na lista!\n");
322:     system("pause");
323: }
324:
325: void imprimirElemento(LISTA *lista, TIPOCHAVE chave){
326:     int i=0, verifica=0;
327:     system("cls");
328:     for(i=0;i<lista->nroElem;i++){
329:         if(chave==lista->vetor[i].chave){
330:             printf("Registro foi encontrado na posicao: %d\n");
331:             verifica=1;
332:         }
333:     }
334:     if(verifica==0)
335:         printf("Registro nao encontrado na lista!\n");
336:     system("pause");
337: }
338:
339: void imprimirListaInicioFim(LISTA *lista){
340:     int i;
341:     if(lista->nroElem==0)
342:         printf("Lista Vazia!\n");
343:     else{
344:         printf("Registros: (POSICAO - REGISTRO)\n");
345:         for(i=0; i<lista->nroElem;i++)
346:             printf("%d - %d\n",i,lista->vetor[i].chave);
347:     }
348:     system("pause");
349: }
350:
351: void imprimirListaFimInicio(LISTA *lista){
352:     int i;
353:     if(lista->nroElem==0)
354:         printf("Lista Vazia!\n");
355:     else{
356:         printf("Registros: (POSICAO - REGISTRO)\n");
357:         for(i=lista->nroElem-1;i>=0;i--)
358:             printf("%d - %d\n",i,lista->vetor[i].chave);
359:     }
360:     system("pause");

```

```
361: }
362:
363: void deletarPosicao(LISTA *lista, int posicao){
364:     int i = 0;
365:     system("cls");
366:     if(posicao<lista->nroElem){
367:         printf("Sucesso!\n");
368:         if(posicao==(lista->nroElem-1))
369:             lista->nroElem = lista->nroElem-1;
370:         else{
371:             for(i=posicao;i<lista->nroElem-1;i++)
372:                 lista->vetor[i] = lista->vetor[i+1];
373:             lista->nroElem = lista->nroElem-1;
374:         }
375:     }else
376:         printf("Posicao nao existe!\n");
377:     system("pause");
378: }
379:
```