

```
1  /*
2  * Proyecto: Polimorfismo-Fichas_Ajedrez
3  * Archivo:  Ficha.h
4  * Autor:    J. Miguel Guanira E.
5  *
6  * Created on 3 de noviembre de 2024, 10:26 PM
7  */
8
9
10 #ifndef FICHA_H
11 #define FICHA_H
12
13 class Ficha {
14 private:
15     char *id;
16     int fila;
17     int columna;
18 public:
19     Ficha();
20     Ficha(const Ficha& orig);
21     virtual ~Ficha();
22     void SetCol(int col);
23     int GetCol() const;
24     void SetFil(int fil);
25     int GetFil() const;
26     void SetId(const char *id);
27     void GetId(char *id) const;
28     void operator = (const Ficha& orig);
29     void asigna(char*id, int fil, int col);
30     virtual void imprime(ofstream &arch) const;
31     virtual void mover(char direccion, int cantidad)= 0;
32 };
33
34 #endif /* FICHA_H */
35
36 /*
37 * Proyecto: Polimorfismo-Fichas_Ajedrez
38 * Archivo:  Ficha.cpp
39 * Autor:    J. Miguel Guanira E.
40 *
41 * Created on 3 de noviembre de 2024, 10:26 PM
42 */
43
44 #include <iostream>
45 #include <fstream>
46 #include <iomanip>
47 using namespace std;
48 #include <cstring>
49 #include "Ficha.h"
50
51 Ficha::Ficha() {
52     id = nullptr;
53 }
54
55 Ficha::Ficha(const Ficha& orig) {
56     id = nullptr;
57     *this = orig;
58 }
59
60 Ficha::~Ficha() {
61     if(id != nullptr) delete id;
62 }
63
64 void Ficha::SetCol(int col) {
65     this->columna = col;
66 }
```

```
67
68     int Ficha::GetCol() const {
69         return columna;
70     }
71
72     void Ficha::SetFil(int fil) {
73         this->fila = fil;
74     }
75
76     int Ficha::GetFil() const {
77         return fila;
78     }
79
80     void Ficha::SetId(const char* ident) {
81         if(id!=nullptr) delete id;
82         id = new char[strlen(ident)+1];
83         strcpy(id, ident);
84     }
85
86     void Ficha::GetId(char*ident) const {
87         if (id == nullptr) ident[0] = 0;
88         else strcpy(ident, id);
89     }
90
91     void Ficha::operator =(const Ficha& orig) {
92         char cad[20];
93         orig.GetId(cad);
94         SetId(cad);
95         fila = orig.fila;
96         columna = orig.columna;
97     }
98
99     void Ficha::asigna(char *ident, int fil, int col) {
100         SetId(ident);
101         SetFil(fil);
102         SetCol(col);
103     }
104
105     void Ficha::imprime(ofstream &arch) const {
106         char ident[10];
107         GetId(ident);
108         arch<<left<<"ID: "<<setw(6)<<ident<<left<<"Fila:"<<right<<setw(3)<<fila
109             <<left<<"  Columna:"<<right<<setw(3)<<columna<<endl;
110     }
111
112     /*
113     * Proyecto: Polimorfismo-Fichas_Ajedrez
114     * Archivo: Torre.h
115     * Autor: J. Miguel Guanira E.
116     *
117     * Created on 5 de noviembre de 2024, 06:04 PM
118     */
119
120
121     #ifndef TORRE_H
122     #define TORRE_H
123     #include "Ficha.h"
124     class Torre: public Ficha {
125     private:
126
127     public:
128         void imprime(ofstream &arch) const;
129         void mover(char direccion, int cantidad);
130     };
131
132     #endif /* TORRE_H */
```

```
133
134  /*
135   * Proyecto: Polimorfismo-Fichas_Ajedrez
136   * Archivo:  Torre.cpp
137   * Autor:    J. Miguel Guanira E.
138   *
139   * Created on 5 de noviembre de 2024, 06:04 PM
140   */
141
142  #include <iostream>
143  #include <fstream>
144  #include <iomanip>
145  using namespace std;
146
147  #include "Torre.h"
148  #include "Ficha.h"
149
150  void Torre::imprime(ofstream& arch) const {
151      arch<<left<<"TORRE    ";
152      Ficha::imprime(arch);
153  }
154
155  void Torre::mover(char direccion, int cantidad) {
156      switch (direccion){
157          case 'P': // ARRIBA
158              case 'p':
159                  SetFil(GetFil()-cantidad);
160                  break;
161          case 'Q': // DERECHA
162              case 'q':
163                  SetCol(GetCol()+cantidad);
164                  break;
165          case 'R': // ABAJO
166              case 'r':
167                  SetFil(GetFil()+cantidad);
168                  break;
169          case 'S': // IZQUIERDA
170              case 's':
171                  SetCol(GetCol()-cantidad);
172      }
173  }
174
175  /*
176   * Proyecto: Polimorfismo-Fichas_Ajedrez
177   * Archivo:  Alfil.h
178   * Autor:    J. Miguel Guanira E.
179   *
180   * Created on 5 de noviembre de 2024, 06:12 PM
181   */
182
183
184  #ifndef ALFIL_H
185  #define ALFIL_H
186  #include "Ficha.h"
187  class Alfil: public Ficha {
188  private:
189
190  public:
191      void imprime(ofstream &arch) const;
192      void mover(char direccion, int cantidad);
193  };
194
195  #endif /* ALFIL_H */
196
197  /*
198   * Proyecto: Polimorfismo-Fichas_Ajedrez
```

```
199  * Archivo:  Alfil.cpp
200  * Autor:    J. Miguel Guanira E.
201  *
202  * Created on 5 de noviembre de 2024, 06:12 PM
203  */
204
205  #include <iostream>
206  #include <fstream>
207  #include <iomanip>
208  using namespace std;
209  #include "Alfil.h"
210
211  void Alfil::imprime(ofstream& arch) const {
212      arch<<left<<"ALFIL    ";
213      Ficha::imprime(arch);
214  }
215
216  void Alfil::mover(char direccion, int cantidad) {
217      switch (direccion){
218          case 'P': // ARRIBA - IZQUIERDA
219              case 'p':
220                  SetFil(GetFil()-cantidad);
221                  SetCol(GetCol()-cantidad);
222                  break;
223          case 'Q': // ARRIBA - DERECHA
224              case 'q':
225                  SetFil(GetFil()-cantidad);
226                  SetCol(GetCol()+cantidad);
227                  break;
228          case 'R': // ABAJO - DERECHA
229              case 'r':
230                  SetFil(GetFil()+cantidad);
231                  SetCol(GetCol()+cantidad);
232                  break;
233          case 'S': // ABAJO - IZQUIERDA
234              case 's':
235                  SetFil(GetFil()+cantidad);
236                  SetCol(GetCol()-cantidad);
237      }
238  }
239
240  /*
241  * Proyecto: Polimorfismo-Fichas_Ajedrez
242  * Archivo:  Caballo.h
243  * Autor:    J. Miguel Guanira E.
244  *
245  * Created on 5 de noviembre de 2024, 06:18 PM
246  */
247
248
249  #ifndef CABALLO_H
250  #define CABALLO_H
251  #include "Ficha.h"
252
253  class Caballo: public Ficha {
254  private:
255
256  public:
257      void imprime(ofstream &arch) const;
258      void mover(char direccion, int cantidad);
259  };
260
261  #endif /* CABALLO_H */
262
263  /*
264  * Proyecto: Polimorfismo-Fichas_Ajedrez
```

```
265  * Archivo:  Caballo.cpp
266  * Autor:    J. Miguel Guanira E.
267  *
268  * Created on 5 de noviembre de 2024, 06:18 PM
269  */
270
271  #include <iostream>
272  #include <fstream>
273  #include <iomanip>
274  using namespace std;
275
276  #include "Caballo.h"
277
278  void Caballo::imprime(ofstream& arch) const {
279      arch<<left<<"CABALLO  ";
280      Ficha::imprime(arch);
281  }
282
283  void Caballo::mover(char direccion, int cantidad) {
284      switch (direccion){
285          case 'P': // ARRIBA - DERECHA
286              SetFil(GetFil()-2);
287              SetCol(GetCol()+1);
288              break;
289          case 'p': // ARRIBA - IZQUIERDA
290              SetFil(GetFil()-2);
291              SetCol(GetCol()-1);
292              break;
293          case 'Q': // DERECHA - ABAJO
294              SetCol(GetCol()+2);
295              SetFil(GetFil()+1);
296              break;
297          case 'q': // DERECHA - ARRIBA
298              SetCol(GetCol()+2);
299              SetFil(GetFil()-1);
300              break;
301          case 'R': // ABAJO - IZQUIERDA
302              SetCol(GetCol()+2);
303              SetFil(GetFil()-1);
304              break;
305          case 'r': // ABAJO - DERECHA
306              SetCol(GetCol()+2);
307              SetFil(GetFil()+1);
308              break;
309          case 'S': // IZQUIERDA - ARRIBA
310              SetCol(GetCol()-2);
311              SetFil(GetFil()-1);
312              break;
313          case 's': // IZQUIERDA - ABAJO
314              SetCol(GetCol()-2);
315              SetFil(GetFil()+1);
316      }
317  }
318
319  /*
320  * Proyecto: Polimorfismo-Fichas_Ajedrez
321  * Archivo:  Tablero.h
322  * Autor:    J. Miguel Guanira E.
323  *
324  * Created on 5 de noviembre de 2024, 06:36 PM
325  */
326
327
328  #ifndef TABLERO_H
329  #define TABLERO_H
330  #include <fstream>
```

```
331 #include "Ficha.h"
332 class Tablero {
333 private:
334     class Ficha *ficha[100];
335     int numFichas;
336 public:
337     Tablero();
338     virtual ~Tablero();
339     void leerFichas(ifstream &);
340     void colocarFicha(char, char*, int, int);
341     void leerMovimientos(ifstream &);
342     void moverFicha(char*, char, int);
343     void imprimeDatos(ofstream &);
344 };
345
346 #endif /* TABLERO_H */
347
348 /*
349  * Proyecto: Polimorfismo-Fichas_Ajedrez
350  * Archivo:  Tablero.cpp
351  * Autor:    J. Miguel Guanira E.
352  *
353  * Created on 5 de noviembre de 2024, 06:36 PM
354  */
355
356 #include <iostream>
357 #include <fstream>
358 #include <iomanip>
359 using namespace std;
360 #include <cstring>
361 #include "Torre.h"
362 #include "Alfil.h"
363 #include "Caballo.h"
364 #include "Tablero.h"
365
366 Tablero::Tablero() {
367     numFichas = 0;
368 }
369
370 Tablero::~Tablero() {
371     for(int i=0; i<numFichas; i++)
372         delete ficha[i];
373 }
374
375 void Tablero::leerFichas(ifstream&arch) {
376     char id[10], tipo;
377     int fil,col;
378     while (1){
379         arch >> tipo;
380         if (arch.eof())break;
381         arch >> id >> fil >> col;
382         colocarFicha(tipo, id, fil, col);
383     }
384 }
385
386 void Tablero::colocarFicha(char tipo, char*id, int fil, int col) {
387     switch(tipo){
388     case 'T':
389         ficha[numFichas] = new class Torre ;
390         break;
391     case 'A':
392         ficha[numFichas] = new Alfil;
393         break;
394     case 'C':
395         ficha[numFichas] = new Caballo;
396     }
```

```
397 // Aquí NO ejecutamos el polimorfismo:
398 ficha[numFichas]->asigna(id,fil,col);
399 numFichas++;
400 }
401
402 void Tablero::leerMovimientos(ifstream&archMov) {
403     char id[10], dir;
404     int cant;
405     while(1){
406         archMov >> id;
407         if(archMov.eof()) break;
408         archMov >> dir >> cant;
409         moverFicha(id, dir, cant);
410     }
411 }
412
413 void Tablero::moverFicha(char *id, char dir, int cant) {
414     char idTab[10];
415     for (int i=0; i<numFichas; i++){
416         ficha[i]->GetId(idTab);
417         if (strcmp(idTab, id)==0) {
418             // Aplicamos el polimorfismo
419             ficha[i]->mover(dir, cant);
420             return;
421         }
422     }
423 }
424
425 void Tablero::imprimeDatos(ofstream&archRep) {
426     for (int i=0; i<numFichas; i++){
427         // Aplicamos el polimorfismo
428         ficha[i]->imprime(archRep);
429     }
430 }
431
432 /*
433  * Proyecto: Polimorfismo-Fichas_Ajedrez
434  * Archivo: MovimientosEnTablero.h
435  * Autor: J. Miguel Guanira E.
436  *
437  * Created on 5 de noviembre de 2024, 07:29 PM
438  */
439
440
441 #ifndef MOVIMIENTOSENTABLERO_H
442 #define MOVIMIENTOSENTABLERO_H
443 #include "Tablero.h"
444
445 class MovimientosEnTablero {
446 private:
447     class Tablero tablero;
448 public:
449     void cargar_Y_MoverFichas(const char*, const char*,
450                             const char*);
451 };
452
453 #endif /* MOVIMIENTOSENTABLERO_H */
454
455 /*
456  * Proyecto: Polimorfismo-Fichas_Ajedrez
457  * Archivo: MovimientosEnTablero.cpp
458  * Autor: J. Miguel Guanira E.
459  *
460  * Created on 5 de noviembre de 2024, 07:29 PM
461  */
462
```

```
463 #include <iostream>
464 #include <iomanip>
465 using namespace std;
466
467 #include "MovimientosEnTablero.h"
468
469 void MovimientosEnTablero::cargar_Y_MoverFichas (
470     const char* nombArchFichas,
471     const char* nombArchMovimientos,
472     const char* nombArchReporte) {
473     ifstream archFichas(nombArchFichas, ios::in);
474     if (!archFichas){
475         cout << "No se pude abrir el archivo: "<<nombArchFichas<<endl;
476         exit(1);
477     }
478
479     ifstream archMovimientos(nombArchMovimientos, ios::in);
480     if (!archMovimientos){
481         cout << "No se pude abrir el archivo: "<<nombArchReporte<<endl;
482         exit(1);
483     }
484     ofstream archReporte(nombArchReporte, ios::out);
485     if (!archReporte){
486         cout << "No se pude abrir el archivo: "<<nombArchReporte<<endl;
487         exit(1);
488     }
489
490     tablero.leerFichas(archFichas);
491     archReporte << "Posiciones iniciales de las fichas:" << endl;
492     tablero.imprimeDatos(archReporte);
493     archReporte << endl << endl;
494     tablero.leerMovimientos(archMovimientos);
495     archReporte << "Posiciones finales: " << endl;
496     tablero.imprimeDatos(archReporte);
497 }
498
499 /*
500 * Proyecto: Polimorfismo-Fichas_Ajedrez
501 * Archivo:  main.cpp
502 * Autor:    J. Miguel Guanira E.
503 *
504 * Created on 3 de noviembre de 2024, 10:25 PM
505 */
506
507 #include <iostream>
508 #include <iomanip>
509 using namespace std;
510 #include "MovimientosEnTablero.h"
511 int main(int argc, char** argv) {
512     class MovimientosEnTablero movEnTabl;
513     movEnTabl.cargar_Y_MoverFichas("Fichas.txt", "Movimientos.txt",
514                                     "Reporte.txt");
515     return 0;
516 }
517
```