

```
1  /*
2  * Proyecto: UsoDEQsortDeCstdlib
3  * Archivo:  main.cpp
4  * Autor:    J. Miguel Guanira E.//miguel.guanira.
5  *
6  * Created on 24 de septiembre de 2024, 09:05 AM
7  */
8
9  #include <iostream>
10 #include <iomanip>
11 using namespace std;
12 #include <cstdlib>
13 // #include "FuncionesDeEnteros.h"
14 #include "FuncionesDeCadenas.h"
15 #include "FuncionesDePunterosGenericos.h"
16 int main(int argc, char** argv) {
17     // int a[50]{10,25, 7,15, 8, 33, 45 , 1, 19, 10,6, 16, 41},n=13;
18     //
19     // qsort(a,n,sizeof(int),miStrInt);
20     // for (int i = 0; i < n; i++)
21     //     cout<<setw(4)<<a[i];
22     // cout<<endl;
23
24     // char **personas;
25     // int nc;
26     // leerDatos(personas,nc,"Personas.txt");
27     // qsort(personas,nc,sizeof(char*),miStrcmp);
28     // for (int i = 0; i < nc; i++)
29     //     cout<<personas[i]<<endl;
30
31     void *personal;
32     int np;
33     leerDatos(personal, np,"personal.csv");
34     imprimirDatos(personal,np,"ReportePersonal.txt");
35     return 0;
36 }
37
38 /*
39 * Proyecto: UsoDEQsortDeCstdlib
40 * Archivo:  FuncionesDeEnteros.h
41 * Autor:    J. Miguel Guanira E. //miguel.guanira.
42 *
43 * Created on 24 de septiembre de 2024, 09:07 AM
44 */
45
46 #ifndef FUNCIONESDEENTEROS_H
47 #define FUNCIONESDEENTEROS_H
48
49 int miStrInt(const void *, const void *);
50
51 #endif /* FUNCIONESDEENTEROS_H */
52
53 /*
54 * Proyecto: UsoDEQsortDeCstdlib
55 * Archivo:  FuncionesDeEnteros.cpp
56 * Autor:    J. Miguel Gunira E//miguel.guanira.
57 *
58 * Created on 24 de septiembre de 2024, 09:07 AM
59 */
60
61 #include <iostream>
62 #include <iomanip>
63 using namespace std;
64
65 int miStrInt(const void *a, const void *b){
66     int *ai = (int *)a, *bi= (int *)b;
```

```
67     return *bi-*ai;
68 }
69
70 /*
71  * Proyecto: UsoDEQsortDeCstdlib
72  * Archivo:  FuncionesDeCadenas.h
73  * Autor:    J. Miguel Guanira E. //miguel.guanira.
74  *
75  * Created on 24 de septiembre de 2024, 09:13 AM
76  */
77
78 #ifndef FUNCIONESDECADENAS_H
79 #define FUNCIONESDECADENAS_H
80
81 void leerDatos(char **&personas,int &nc, const char*);
82 char*leeCadena(istream &arch, char delimitador='\n');
83 int miStrcmp(const void*, const void *);
84
85 #endif /* FUNCIONESDECADENAS_H */
86
87 /*
88  * Proyecto: UsoDEQsortDeCstdlib
89  * Archivo:  FuncionesDeCadenas.cpp
90  * Autor:    J. Miguel Gunira E//miguel.guanira.
91  *
92  * Created on 24 de septiembre de 2024, 09:13 AM
93  */
94
95 #include <iostream>
96 #include <fstream>
97 #include <iomanip>
98 using namespace std;
99 #include <cstring>
100 #include "FuncionesDeCadenas.h"
101
102 void leerDatos(char **&personas,int &nc, const char*nombArch){
103     char*buff[200];
104     nc=0;
105     ifstream arch(nombArch,ios::in);
106     if(not arch.is_open()){
107         cout<<"ERROR: No se pudo abrir el archivo "<<nombArch<<endl;
108         exit(1);
109     }
110     while(true){
111         buff[nc] = leeCadena(arch);
112         if(arch.eof())break;
113         nc++;
114     }
115     personas = new char*[nc];
116     for (int i = 0; i < nc; i++)
117         personas[i] = buff[i];
118 }
119
120 char*leeCadena(istream &arch, char delimitador){
121     char buffer[60], *cad;
122     arch.getline(buffer,60,delimitador);
123     cad = new char[strlen(buffer)+1];
124     strcpy(cad, buffer);
125     return cad;
126 }
127
128 int miStrcmp(const void*a, const void *b){
129     char **ai = (char**)a,**bi = (char**)b;
130     return -strcmp(*ai,*bi);
131 }
132
```

```
133  /*
134  * Proyecto: UsoDEQsortDeCstdlib
135  * Archivo:  FuncionesDePunterosGenericos.h
136  * Autor:    J. Miguel Guanira E. //miguel.guanira.
137  *
138  * Created on 24 de septiembre de 2024, 09:34 AM
139  */
140
141  #ifndef FUNCIONESDEPUNTEROSGENERICOS_H
142  #define FUNCIONESDEPUNTEROSGENERICOS_H
143  void leerDatos(void*&per, int &numDat, const char *nombArch);
144  void *leerRegistro(istream &arch);
145  void aumentamosEspacios(void **&personal, int &numDat, int &cap);
146  void imprimirDatos(void*per, int np, const char *nombArch);
147  void imprimePersona(ofstream &arch, void *per);
148
149  #endif /* FUNCIONESDEPUNTEROSGENERICOS_H */
150
151  /*
152  * Proyecto: UsoDEQsortDeCstdlib
153  * Archivo:  FuncionesDePunterosGenericos.cpp
154  * Autor:    J. Miguel Gunira E//miguel.guanira.
155  *
156  * Created on 24 de septiembre de 2024, 09:34 AM
157  */
158
159  #include <iostream>
160  #include <fstream>
161  #include <iomanip>
162  using namespace std;
163  #include <cstring>
164  #include "FuncionesDeCadenas.h"
165  #include "FuncionesDePunterosGenericos.h"
166  #define INCREMENTO 5
167
168  void leerDatos(void*&per, int &numDat, const char *nombArch){
169      ifstream arch(nombArch, ios::in);
170      if(not arch.is_open()){
171          cout<<"ERROR: No se pudo abrir el archivo "<<nombArch<<endl;
172          exit(1);
173      }
174      void **personal, *reg;
175      int cap=0;
176      numDat =0;
177      personal = nullptr;
178      while(true){
179          reg = leerRegistro(arch);
180          if(arch.eof())break;
181          if(numDat == cap) aumentamosEspacios(personal, numDat, cap);
182          personal[numDat-1] = reg;
183          numDat++;
184      }
185      per = personal;
186  }
187
188  void *leerRegistro(istream &arch){
189      void **registro;
190      int *codigo, cod;
191      char *nombre;
192      double *sueldo;
193      arch >> cod;
194      if(arch.eof()) return nullptr;
195      codigo = new int;
196      *codigo = cod;
197      arch.get();
198      nombre = leeCadena(arch, ',');
```

```
199     sueldo = new double;
200     arch>>*sueldo;
201     registro = new void*[3]{};
202     registro[0] = codigo;
203     registro[1] = nombre;
204     registro[2] = sueldo;
205     return registro;
206 }
207
208
209 void aumentamosEspacios(void **&personal,int &numDat,int &cap){
210     void **aux;
211     cap += INCREMENTO;
212     if(personal == nullptr){
213         personal = new void*[cap]{};
214         numDat = 1;
215     }
216     else{
217         aux =new void*[cap]{};
218         for (int i = 0; i < numDat; i++) {
219             aux[i] = personal[i];
220         }
221         delete personal;
222         personal = aux;
223     }
224 }
225
226 void imprimirDatos(void*per, int np, const char *nombArch){
227     ofstream arch(nombArch,ios::out);
228     if(not arch.is_open()){
229         cout<<"ERROR: No se pudo abrir el archivo "<<nombArch<<endl;
230         exit(1);
231     }
232     void **personal = (void **)per;
233     for (int i = 0; i<np-1 ; i++)
234         imprimePersona(arch,personal[i]);
235 }
236
237 void imprimePersona(ofstream &arch,void *per){
238     void **persona = (void **)per;
239     int *codigo = (int *)persona[0];
240     char*nombre = (char*)persona[1];
241     double *sueldo = (double*)persona[2];
242
243     arch.precision(2);
244     arch<<fixed;
245     arch<<right<<setw(10)<<*codigo<<" " <<left<<nombre<<right
246         <<setw(10)<<*sueldo <<endl;
247
248 }
```