

```
1  /*
2  * Proyecto: UsoDeQsortDeCstdlib
3  * Archivo:  main.cpp
4  * Autor:    J. Miguel Guanira E.//miguel.guanira.
5  *
6  * Created on 17 de abril de 2024, 09:07 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 "FuncionesDePunterosVoid.h"
16
17 int main(int argc, char** argv) {
18     // int a[50]={10, 15, 2, 82, 35, 44, 17, 91,33, 77,12, 21, 56,17, 4, 25}, nd=16;
19     // qsort(a,nd,sizeof(int),intcmp);
20     // for(int i=0; i<nd; i++)
21     //     cout<<setw(4)<<a[i];
22     // cout<<endl;
23
24     // // Con cadenas
25     // int nd;
26     // char *nombres[100];
27     // leerDatos(nombres,nd, "Personas.txt");
28     // qsort(nombres,nd,sizeof(char*),cadcmp);
29     // for(int i=0; i<nd; i++)
30     //     cout<<nombres[i]<<endl;
31
32     void *personal;
33     int nd;
34     leerDatos(personal,nd,"personal.csv");
35     qsort(personal,nd, sizeof(void*),voidcmpSueldo);
36     // qsort(personal,nd, sizeof(void*),voidcmpCodigos);
37     // qsort(personal,nd, sizeof(void*),voidcmpNombres);
38     imprimirDatos(personal,nd);
39
40     return 0;
41 }
42
43 /*
44 * Proyecto: UsoDeQsortDeCstdlib
45 * Archivo:  FuncionesDeEnteros.h
46 * Autor:    J. Miguel Guanira E. //miguel.guanira.
47 *
48 * Created on 17 de abril de 2024, 09:09 AM
49 */
50
51 #ifndef FUNCIONESDEENTEROS_H
52 #define FUNCIONESDEENTEROS_H
53
54 int intcmp(const void*a, const void*b);
55
56 #endif /* FUNCIONESDEENTEROS_H */
57
58 /*
59 * Proyecto: UsoDeQsortDeCstdlib
60 * Archivo:  FuncionesDeEnteros.cpp
61 * Autor:    J. Miguel Gunira E//miguel.guanira.
62 *
63 * Created on 17 de abril de 2024, 09:09 AM
64 */
65
66 #include <iostream>
```

```
67  #include <iomanip>
68  using namespace std;
69
70  int intcmp(const void*a, const void*b){
71      int*ai=(int*)a, *bi=(int*)b;
72      return *ai-*bi;
73  }
74
75  /*
76   * Proyecto: UsoDeQsortDeCstdlib
77   * Archivo:  FuncionesDeCadenas.h
78   * Autor:    J. Miguel Guanira E. //miguel.guanira.
79   *
80   * Created on 17 de abril de 2024, 09:24 AM
81   */
82
83  #ifndef FUNCIONESDECADENAS_H
84  #define FUNCIONESDECADENAS_H
85
86  void leerDatos(char**nombres,int &nd, const char*nombArch);
87  int cadcmp(const void*cad1,const void*cad2);
88
89  #endif /* FUNCIONESDECADENAS_H */
90
91  *
92  * Proyecto: UsoDeQsortDeCstdlib
93  * Archivo:  FuncionesDeCadenas.cpp
94  * Autor:    J. Miguel Gunira E//miguel.guanira.
95  *
96  * Created on 17 de abril de 2024, 09:24 AM
97  */
98
99  #include <iostream>
100 #include <fstream>
101 #include <iomanip>
102 using namespace std;
103 #include <cstring>
104
105 void leerDatos(char**nombres,int &nd, const char*nombArch){
106     ifstream arch(nombArch,ios::in);
107     if(not arch.is_open()){
108         cout<<"ERROR: No se pudo abrir el archivo "<<nombArch<<endl;
109         exit(1);
110     }
111     char nomb[60];
112     nd=0;
113
114     while(true){
115         arch>>nomb;
116         if(arch.eof())break;
117         nombres[nd] = new char[strlen(nomb)+1];
118         strcpy(nombres[nd],nomb);
119         nd++;
120     }
121 }
122
123 int cadcmp(const void*cad1,const void*cad2){
124     char **cad1I = (char**)cad1, **cad2I = (char**)cad2;
125
126     return strcmp(*cad1I,*cad2I);
127 //    return strcmp(*cad2I,*cad1I);
128 }
129
130 /*
131 * Proyecto: UsoDeQsortDeCstdlib
132 * Archivo:  FuncionesDePunterosVoid.h
```

```

133  * Autor:      J. Miguel Guanira E. //miguel.guanira.
134  *
135  * Created on 17 de abril de 2024, 09:44 AM
136  */
137
138  #ifndef FUNCIONESDEPUNTEROSVOID_H
139  #define FUNCIONESDEPUNTEROSVOID_H
140
141  int voidcmpSueldo(const void*a, const void*b);
142  void leerDatos(void *&persona,int &numDat,const char*nombArch);
143  void *leerReg(istream&arch);
144  void imprimirDatos(void *persona ,int numDat);
145  void imprimeRegistro(void*reg);
146  int voidcmpCodigos(const void *a, const void *b);
147  int voidcmpNombres(const void *a, const void *b);
148
149  #endif /* FUNCIONESDEPUNTEROSVOID_H */
150
151  /*
152  * Proyecto: UsoDeQsortDeCstdlib
153  * Archivo:  FuncionesDePunterosVoid.cpp
154  * Autor:    J. Miguel Gunira E//miguel.guanira.
155  *
156  * Created on 17 de abril de 2024, 09:44 AM
157  */
158
159  #include <iostream>
160  #include <fstream>
161  #include <iomanip>
162  using namespace std;
163  #include <cstring>
164  #include "FuncionesDePunterosVoid.h"
165
166  enum Registro {CODIGO, NOMBRE, SUELDO};
167
168  void leerDatos(void *&persona,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 *buff[500], **per, *p;
175      numDat = 0;
176      while(1){
177          p = leerReg(arch);
178          if(p == nullptr) break;
179          buff[numDat] = p;
180          numDat++;
181      }
182      per = new void*[numDat];
183      for(int i=0; i<numDat; i++)
184          per[i] = buff[i];
185      persona = per;
186  }
187
188  void *leerReg(istream&arch){
189      void **r;
190      int *codigo, cod;
191      char*nombre, buff[100];
192      double*sueldo;
193
194      arch>>cod;
195      if(arch.eof()) return nullptr;
196      codigo = new int;
197      *codigo = cod;
198      arch.get(); // Sacamos la coma

```

```
199     arch.getline(buff,100,' ');
200     nombre = new char[strlen(buff)+1];
201     strcpy(nombre,buff);
202     sueldo = new double;
203     arch>>*sueldo;
204
205     r = new void*[3];
206     r[CODIGO] = codigo;
207     r[NOMBRE] = nombre;
208     r[SUELDO] = sueldo;
209
210     return r;
211 }
212
213 void imprimirDatos(void *persona ,int numDat){
214     void **per = (void **)persona;
215
216     for(int i=0; i<numDat; i++)
217         imprimeRegistro(per[i]);
218 }
219
220 void imprimeRegistro(void*reg){
221     void **r = (void**)reg;
222     int *codigo = (int *)r[CODIGO];
223     char *nombre = (char*)r[NOMBRE];
224     double *sueldo = (double*)r[SUELDO];
225
226     cout.precision(2);
227     cout<<fixed;
228     cout<<left<<setw(10)<<*codigo<<setw(45)<<nombre
229         <<right<<setw(10)<<*sueldo<<endl;
230 }
231
232 int voidcmpSueldo(const void*a, const void*b){
233     void **ai=(void**)a, **bi=(void**)b;
234     void **regA= (void**) (*ai), **regB=(void**) (*bi);
235     double*sueldoA=(double*) (regA[SUELDO]), *sueldoB= (double*) (regB[SUELDO]);
236     return *sueldoB-*sueldoA;
237 }
238
239 int voidcmpCodigos(const void *a, const void *b){
240     void **ai=(void**)a, **bi=(void**)b;
241     void **regA = (void**) (ai[0]), **regB = (void**) (*bi);
242     int *codA = (int *) (regA[CODIGO]), *codB=(int *) (regB[CODIGO]);
243     return *codA-*codB;
244 }
245
246
247 int voidcmpNombres(const void *a, const void *b){
248     void **ai=(void**)a, **bi=(void**)b;
249     void **regA = (void**) (ai[0]), **regB = (void**) (*bi);
250     char *nomA = (char *) (regA[NOMBRE]), *nomB=(char *) (regB[NOMBRE]);
251     return strcmp(nomA,nomB);
252 }
253
```