

```
1 # Materia.s
2 #
3 # Estructura de datos que implementa el TAD
4 # Materia.
5 #
6 # codigo:      String con el codigo de la materia.
7 # nombre:      String con el nombre de la materia.
8 # creditos:    Entero con los creditos que vale la materia.
9 # cupos:       Entero con el los cupos restantes de la materia.
10 # minCreditos: Entero con el minimo de creditos aprobados
11 #              necesarios para inscribir la materia.
12 # estudiantes: Lista de Pares <Estudiante, Operacion> que
13 #              solicitaron inscripcion a la materia.
14 #
15 # Autores: Ka Fung 18-10492 & Christopher Gomez 18-10892
16 # Fecha: 25-nov-2021
17
18     .data
19
20     .text
21
22 # Funcion crear.
23 # Crea un Materia con los parametros dados.
24 # Entrada:  $a0: codigo.
25 #           $a1: nombre.
26 #           $a2: credito.
27 #           $a3: cupos.
28 #           4($fp): minCreditos.
29 # Salida:   $v0: Materia (negativo si no se pudo crear).
30 #           ($v0): codigo.
31 #           4($v0): nombre.
32 #           8($v0): creditos.
33 #           12($v0): cupos.
34 #           16($v0): minCreditos.
35 #           20($v0): estudiantes.
36 #
37 # Planificacion de registros:
38 # $s0: carne del estudiante.
39 # $s1: Direccion de la materia
40 Materia_crear:
41     # Prologo
42     sw    $fp,    ($sp)
43     sw    $ra,    -4($sp)
44     sw    $s0,    -8($sp)
45     sw    $s1,    -12($sp)
46     move  $fp,    $sp
47     addi  $sp,    $sp, -16
48
49     # Guarda el valor de $a0
50     move  $s0,    $a0
51
52     # Asigna memoria para la Materia.
53     li    $a0,    24
54     li    $v0,    9
55     syscall
56
57     bltz  $v0,    Materia_crear_fin
```

```
58     move $s1, $v0
59
60     # Crea la lista vacia de estudiantes
61     jal Lista_crear
62     bltz $v0, Materia_crear_fin
63
64     # Inicializa la Materia.
65     sw $s0, ($s1) # codigo
66     sw $a1, 4($s1) # nombre
67     sw $a2, 8($s1) # creditos
68     sw $a3, 12($s1) # cupos
69
70     lw $a0, 4($fp)
71     sw $a0, 16($s1) # minCreditos
72     sw $v0, 20($s1) # Lista de estudiantes
73
74     move $v0, $s1
75 Materia_crear_fin:
76     # Epilogo
77     move $sp, $fp
78     lw $fp, ($sp)
79     lw $ra, -4($sp)
80     lw $s0, -8($sp)
81     lw $s1, -12($sp)
82
83     jr $ra
84
85 # Funcion aumentarCupo.
86 # Suma uno al numero de cupos de la materia.
87 # Entrada:  $a0: Materia.
88 #
89 # Planificacion de registros:
90 # $t0: Cupos de la materia.
91 Materia_aumentarCupo:
92     # Prologo
93     sw $fp, ($sp)
94     move $fp, $sp
95     addi $sp, $sp, -4
96
97     # Carga los cupos de la materia.
98     lw $t0, 12($a0)
99     addi $t0, $t0, 1
100    sw $t0, 12($a0)
101
102    # Epilogo
103    move $sp, $fp
104    lw $fp, ($sp)
105
106    jr $ra
107
108
109 # Funcion disminuirCupo.
110 # Resta uno al numero de cupos de la materia.
111 # Entrada:  $a0: Materia.
112 #
113 # Planificacion de registros:
114 # $t0: Cupos de la materia.
```

```
115 Materia_disminuirCupo:
116     # Prologo
117     sw    $fp, ($sp)
118     move  $fp, $sp
119     addi  $sp, $sp, -4
120
121     # Carga los cupos de la materia.
122     lw    $t0, 12($a0)
123     addi  $t0, $t0, -1
124     sw    $t0, 12($a0)
125
126     # Epilogo
127     move  $sp, $fp
128     lw    $fp, ($sp)
129
130     jr    $ra
131
132 # Funcion agregarEstudiante
133 # Agrega un estudiante a la lista de estudiantes de la materia.
134 # Entrada:  $a0: Materia.
135 #           $a1: Estudiante.
136 #           $a2: ASCII caracter con la operacion.
137 #
138 # Planificacion de registros:
139 # $s0: Materia.
140 # $t0: #Cupos de la materia
141 Materia_agregarEstudiante:
142     # Prologo
143     sw    $fp, ($sp)
144     move  $fp, $sp
145     sw    $ra, -4($sp)
146     sw    $s0, -8($sp)
147     addi  $sp, $sp, -12
148
149     # Guarda la materia
150     move  $s0, $a0
151
152     # Crea el Par(Estudiante, Op.)
153     move  $a0, $a1 # Estudiante
154     move  $a1, $a2 # Op.
155     jal   Par_crear
156
157     # Verifica la creacion del Par.
158     bltz  $v0, Materia_agregarEstudiante_fin
159     move  $a1, $v0
160
161     # Agrega el Par a la lista de estudiantes.
162     move  $a0, $s0
163     lw    $a0, 20($a0)
164     la    $a2, comparador_carnet
165     jal   Lista_insertarOrdenado
166
167     # Disminuye por uno el número de cupos.
168     lw    $t0, 12($s0)
169     addi  $t0, $t0, -1
170     sw    $t0, 12($s0)
171
```

```
172 Materia_agregarEstudiante_fin:
173     # Epilogo
174     move $sp, $fp
175     lw    $fp, ($sp)
176     lw    $ra, -4($sp)
177     lw    $s0, -8($sp)
178
179     jr $ra
180
181 # Funcion eliminarEstudiante.
182 # Marca un estudiante de la lista de estudiantes de la materia.
183 # como eliminado.
184 # Entrada:  $a0: Materia.
185 #           $a1: Estudiante.
186 #
187 # Planificacion de registros:
188 # $t0: Lista de estudiantes de Materia
189 # $t1: Centinela de la Lista.
190 # $t2: Nodo de la Lista.
191 # $t3: Valor del nodo (Par).
192 # $t4: Estudiante del Par.
193 # $t5: Auxiliar.
194 Materia_eliminarEstudiante:
195     # Prologo
196     sw    $fp, ($sp)
197     move $fp, $sp
198     addi $sp, $sp, -4
199
200     lw $t0, 20($a0) # Lista estudiantes de Materia
201     lw $t1, ($t0)   # Centinela de la lista
202     lw $t2, 8($t1)  # Primer nodo de la lista
203
204     for_est_mat:
205         # while Nodo != centinela
206         beq $t1, $t2, for_est_mat_fin
207
208         lw $t3, 4($t2) # Valor del nodo (Par)
209         lw $t4, ($t3)  # Estudiante del par
210
211         # Si par.Estudiante == solicitud.Estudiante
212         # modificamos la operacion a 'E'
213         beq $t4, $a1, eliminar_operacion
214
215         # Actualizamos al Nodo.siguiente
216         lw $t2, 8($t2)
217         b for_est_mat
218
219     eliminar_operacion:
220         li $t5, 69
221         sb $t5, 4($t3)
222
223     for_est_mat_fin:
224         # Disminuye por uno el número de cupos.
225         lw $t5, 12($a0)
226         addi $t5, $t5, 1
227         sw $t5, 12($a0)
228
```

```
229 Materia_eliminarEstudiante_fin:
230     # Epilogo
231     move $sp, $fp
232     lw    $fp, ($sp)
233
234     jr $ra
235
236
237 # Funcion imprimirMateria
238 # Imprime en un archivo la materia.
239 # Entrada:  $a0: Materia.
240 #           $a1: Archivo.
241 #
242 # Planificacion de registros:
243 # $s0: Materia.
244 # $s1: Archivo
245 # $s2: Caracter actual del nombre de materia
246 Materia_imprimirMateria:
247     # Prologo
248     sw    $fp,    ($sp)
249     sw    $ra, -4($sp)
250     sw    $s0, -8($sp)
251     sw    $s1, -12($sp)
252     sw    $s2, -16($sp)
253     move  $fp,    $sp
254     addi  $sp,    $sp, -20
255
256     move  $s0, $a0
257     move  $s1, $a1
258
259     # Imprime Materia.codigo
260     li    $v0, 15
261     move  $a0, $s1
262     lw    $a1, ($s0)
263     li    $a2, 7
264     syscall
265
266     # Imprime ' "'
267     li    $v0, 15
268     move  $a0, $s1
269     la    $a1, espC
270     li    $a2, 2
271     syscall
272
273     # Imprime Materia.nombre
274     lw    $s2, 4($s0)
275     for_letra:
276         lb    $a1, ($s2)
277         beqz  $a1, for_letra_fin
278
279         li    $v0, 15
280         move  $a0, $s1
281         move  $a1, $s2
282         li    $a2, 1
283         syscall
284
285         add  $s2, $s2, 1
```

```
286         b for_letra
287
288     for_letra_fin:
289
290     # Imprime ' '
291     li    $v0, 15
292     move  $a0, $s1
293     la    $a1, cEsp
294     li    $a2, 2
295     syscall
296
297     # Imprime Materia.cupos
298     lw    $a0, 12($s0)
299     li    $a1, 3
300
301     jal   itoa
302     move  $a1, $v0
303     move  $a2, $v1
304
305     move  $a0, $s1
306     li    $v0, 15
307     syscall
308
309     # Imprime '\n'
310     li    $v0, 15
311     move  $a0, $s1
312     la    $a1, newl
313     li    $a2, 1
314     syscall
315
316     # Imprime estudiantes
317     lw    $a0, 20($s0)
318     move  $a1, $s1
319     jal   Materia_imprimirEstudiantes
320
321     # Imprime '\n'
322     li    $v0, 15
323     move  $a0, $s1
324     la    $a1, newl
325     li    $a2, 1
326     syscall
327
328 Materia_imprimirMateria_fin:
329     # Epilogo
330     move  $sp,      $fp
331     lw    $fp,      ($sp)
332     lw    $ra,      -4($sp)
333     lw    $s0,      -8($sp)
334     lw    $s1,      -12($sp)
335     lw    $s2,      -16($sp)
336
337     jr    $ra
338
339 # Funcion imprimirEstudiantes.
340 # Imprime en un archivo los estudiantes de una materia.
341 # Entrada:  $a0: Lista de Estudiantes.
342 #           $a1: Archivo.
```

```
343 #
344 # Planificacion de registros:
345 # $t0: Materia.
346 # $t1: Archivo.
347 # $t2: Centinela de la lista.
348 # $t3: Nodo de la lista.
349 # $t4: Valor del nodo (Par).
350 # $t5: Estudiante.
351 # $t6: Caracter del nombre del Estudiante
352 # $t7: Operacion de la solicitud
353 # $t8: ASCII 'S'
354 Materia_imprimirEstudiantes:
355     # Prologo
356     sw    $fp,    ($sp)
357     move  $fp,    $sp
358     addi  $sp,    $sp, -4
359
360     move  $t0, $a0    # Lista estudiantes
361     move  $t1, $a1    # Archivo
362
363     lw    $t2, ($t0)  # Centinela de la lista
364     lw    $t3, 8($t2) # Primer nodo de la lista
365
366     for_imprimir_est:
367         # while Nodo != centinela
368         beq $t2, $t3, Materia_imprimirEstudiantes_fin
369
370         lw $t4, 4($t3) # Valor del nodo (Par)
371         lw $t5, ($t4)  # Estudiante
372
373         # Imprime ' '
374         li $v0, 15
375         move $a0, $t1
376         la  $a1, ident
377         li  $a2, 3
378         syscall
379
380         # Imprime Estudiante.carnet
381         li $v0, 15
382         move $a0, $t1
383         lw  $a1, ($t5)
384         li  $a2, 8
385         syscall
386
387         #Imprime ' "'
388         li $v0, 15
389         move $a0, $t1
390         la  $a1, espC
391         li  $a2, 2
392         syscall
393
394         #Imprime Estudiante.nombre
395         lw $t6, 4($t5)
396         for_letra_est:
397             lb  $a1, ($t6)
398             beqz $a1, for_letra_est_fin
399
```

```
400         li    $v0, 15
401         move  $a0, $t1
402         move  $a1, $t6
403         li    $a2, 1
404         syscall
405
406         add   $t6, $t6, 1
407         b     for_letra_est
408
409     for_letra_est_fin:
410
411     # Imprime ' '
412     li    $v0, 15
413     move  $a0, $t1
414     la    $a1, cEsp
415     li    $a2, 2
416     syscall
417
418     # Imprimir operacion si es necesario
419     lw    $t7, 4($t4)
420     li    $t8, 'S'      # 'S'
421     beq   $t7, $t8, for_imprimir_est_sig
422
423     # Imprime '('
424     li    $v0, 15
425     move  $a0, $t1
426     la    $a1, parentIzq
427     li    $a2, 1
428     syscall
429
430     # Imprime operacion
431     li    $v0, 15
432     move  $a0, $t1
433     la    $a1, 4($t4)
434     li    $a2, 1
435     syscall
436
437     # Imprime ')'
438     li    $v0, 15
439     move  $a0, $t1
440     la    $a1, parentDer
441     li    $a2, 1
442     syscall
443
444     for_imprimir_est_sig:
445     # Imprime '\n'
446     li    $v0, 15
447     move  $a0, $t1
448     la    $a1, newl
449     li    $a2, 1
450     syscall
451
452     # Actualizamos al Nodo.siguiente
453     lw    $t3, 8($t3)
454
455     b     for_imprimir_est
456
```



```
457 Materia_imprimirEstudiantes_fin:
458     # Epilogo
459     move $sp,      $fp
460     lw      $fp,    ($sp)
461
462     jr $ra
```