

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA



Ingeniería en computacion

Programación Estructurada

ALUMNO: Miguel Angel Portillo Attwell

MATRÍCULA: 370097

GRUPO: 432

PROFESOR: Pedro Nuñez Yepiz



Programación Estructurada



```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <time.h>
5 #include "maikylib.h"
6
7 #define MAX 2500
8
9 int msgs();
10 void menu();
11
12 void agg_autom(Talum vect[], int *n, int max);
13 void edit_datos(Talum vect[], int n, int band);
14 void elim_datos(Talum vect[], int n, int band);
15 void busc_datos(Talum vect[], int n, int band);
16 void ord_datos(Talum vect[], int n, int *band);
17 void imp_datos(Talum vect[], int n);
18 void crear_arch_txt(Talum vect[], int n);
19 void ver_arch_txt(Talum vect[], int n);
20 void crear_arch_bin(Talum vect[], int n);
21 void cargar_arch_bin(Talum vect[], int *n, int max, int *band);
22
23 int main()
24 {
25     int msgs();
26     void menu();
27     void agg_autom(Talum vect[], int *n, int max);
28     void edit_datos(Talum vect[], int n, int band);
29     void elim_datos(Talum vect[], int i, int band);
30     void busc_datos(Talum vect[], int n, int band);
31     void ord_datos(Talum vect[], int n, int *band);
32     void imp_datos(Talum vect[], int n);
33     void crear_arch_txt(Talum vect[], int n);
34     void ver_arch_txt(Talum vect[], int n);
35     void crear_arch_bin(Talum vect[], int n);
36     void cargar_arch_bin(Talum vect[], int *n, int max, int *band);
37 }
```



Programación Estructurada

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 #define TRUE 1
6 #define FALSE 0
7 #define NUM_COL 4
8
9 typedef long Tkey;
10
11 > typedef struct _talum{
12 }talum;
13
14 > int vali_int(int ran_inf, int ran_sup, char msg[], char msg_error[])...
15
16 > long vali_long(long ran_inf, long ran_sup, char msg[], char msg_error[])...
17
18 > void vali_num_cad(char cadena[], int ran_inf, int ran_sup, int max, char *msg, char *msg_
19
20 > int busq_sec(int vect[], int len, int val)...
21
22 > char busq_vocal(char cadena[])...
23
24 > char busq_vocal_dela(char cadena[])...
25
26 > char busq_vocal_del(char cadena[])...
27
28 > char busq_cons(char cadena[])...
29
30 > char busq_cons_dela(char cadena[])...
31
32 > char busq_cons_del(char cadena[])...
33
34 > int busq_sec_talum(Talum *alum, int i, long matri)...
35
36 > int busq_bin_talum(Talum *alum, int i, long matri)...
37
38 > void ord_burb(Talum alum[], int i)...
39
40 > void swap(long *a, long *b) ...
41
42 > int partition(Talum alum[], int min, int max)...
43
44 > void ord_quicksort(Talum alum[], int min, int max)...
45
46 > int existe_vect(int num, int *vect, int len)...
47
48 > int existe_talum_matri(long matri, Talum *alum, int i)...
49
50 > int existe_mat(int num, int mat[][NUM_COL], int len)...
51
52 > void uno_a_dos_digitos(char cadena[2])...
53
54 > int comp_esp(char cadena[])...
55
56 > int solo_letras(char cadena[])...
57
58 > void vali_cad(char cadena[], char msg[])...
59
60 > int comp_esp_obg(char cadena[])...
61
62 > int solo_letras_obg(char cadena[])...
63
64 > void vali_cad_obg(char cadena[], char msg[])...
65
66 > void enie(char cadena[])...
67
68 > void enie_mayus(char cadena[])...
69
70 > int dela_del(char ap[])...
71
72 > Talum datos_autom(Talum alum[], int i)...
73
74 > Talum datos_autom_inac(Talum alum[], int i)...
```