

1. Construir una función que permita procesar 7 años (1ero. a 7mo.). Por cada año se tiene e ingresan una cantidad desconocida de números de legajos (mayores a 0) de alumnos(no se sabe qué cantidad de alumnos hay por grado ya que hay alumnos nuevos y otros que pidieron el pase a otra escuela). Finalizan los ingresos de este proceso con legajo 0 (cero) y por cada legajo se ingresa:
 - grado al cual pertenece el alumno, A ó B , tipo caracter, no están ordenados, es decir que indistintamente pueden ingresarse tanto A como B
 - nota, es un número real, de 1 a 10

Esta función debe emitir, por cada grado el promedio de notas por cada división. Ej.:

```

****
(*) Grado 1 – promedio general:
A = 6.89
B = 7.10

*****
(*) Grado 2 – promedio general:
A = 5.90
B = 9.80

```

2. Construir una función que reciba (al menos) grado (A ó B) y nota, y acumule en función de A y B, la suma total de notas por cada uno, contando además la cantidad. La finalidad de esta función sólo es acumular y contar para que la función 1 pueda emitir el mensaje del ejemplo del ítem anterior. En esta función no se ingresan valores por teclado, no retorna un valor, ni se emiten mensajes.

3. Mensaje final en main(), que emita:
 - total de legajos procesados.
 - el mayor promedio de todos, año y grado (A ó B) que lo obtuvo.

Observen -siguiendo los colores- cómo están vinculadas todas las funciones y como una presta servicio a la otra.

Nota aclaratoria: este problema puede resolverse de otras formas, el ejemplo presente es una solución.

main() (3)	Función Proceso_1	Función Proceso_2
Resultados finales	Resultados parciales	Función auxiliar
<pre> int main(){ int tlp = 0, anio; float promedio; char grado; /*invoco la función Proceso1 que retornará la cantidad de legajos procesados*/ tlp=Proceso1(&promedio,&anio,&grado); /*necesito pasar estas variables por dirección para se muestren en main*/ printf("Cantidad de legajos procesados %d\n", tlp); printf("El mayor promedio de todos es %.2f, del grado %c a%co %d", promedio, grado,164,anio); getchar(); return 0; } </pre>	<pre> int Proceso1(float * prom, int * anio, char * grad){ int i,leg,contA, contB, tlp=0; char grado; float nota, acumA,acumB, promA, promB, mayProm=0; for(i = 1; i <= 2; i++){ /* Inicializo contadores y acumuladores porque deben estar en 0 cada vez que cambia el año procesado */ acumA = 0; acumB=0; contA = 0; contB = 0; printf("Ingrese un n%cmerno de legajo: \n",163); scanf("%d",&leg); /* variable de control del while, recordar que no sé cuántos legajos se procesarán*/ fflush(stdin); /*como tengo variables char debo vaciar el buffer del teclado */ while (leg > 0){ printf("Ingrese el grado, A %c B: \n", 162); scanf("%c", &grado); fflush(stdin); </pre>	<pre> void Proceso2(char grado,float nota,float * aA,float * aB,int * cA,int * cB){ switch(grado){ case 'a': case 'A': *aA = *aA + nota; (*cA)++;/*importante colocar los paréntesis*/ break; case 'b': case 'B': *aB = *aB + nota; (*cB)++; break; } } } </pre>

	<pre> printf("Ingrese la nota: "); scanf("%f", &nota); fflush(stdin); Proceso2(grado, nota, &acumA, &acumB, &contA,&contB); printf("Ingrese un n°cmro de legajo, para terminar pulse 0:\n ",163); scanf("%d",&leg); fflush(stdin); } tlp=tlp+contA+contB; /* para este ejemplo suponemos que siempre hay alumnos con notas en todas las gradoisiones */ if (contA > 0) {promA=(float)acumA / contA; printf ("A%co:%d -> promedio A:%.2f\n",164,i, promA); } if (contB > 0) {promB=(float)acumB / contB; printf ("A%co:%d -> promedio B:%.2f\n",164,i, promB); } if(promA > *prom) { *prom = promA; *anio = i; *grad = 'A'; } if(promB > *prom) { *prom = promB; *anio = i; *grad = 'B'; } } return tlp; } </pre>																			
<p>En función main sólo emitimos los resultados finales (no hay procesos). Según la consigna, necesitamos emitir 4 resultados.</p> <ul style="list-style-type: none"> • <u>Dónde realizamos el proceso que nos permite darles valor a esas 4 variables?</u> En la función Proceso_1. • <u>Cómo vinculamos estas variables con una función que es externa a main?</u> Por un lado sabemos que hay funciones que retornan un valor, aquí obtendríamos un resultado. Asignamos el resultado de llamar a Proceso_1 a una variable (tlp). • <u>Cómo resolvemos el resto de las variables, o sea 3 más?</u> sabemos que podemos pasar parámetros por dirección, única forma de no perder los valores cuando Proceso_1 termine su trabajo, y sería nuestra solución. 	<p>En esta función tenemos que realizar el proceso de recorrer 7 años (1 a 7) y por cada año ingresar una cantidad desconocida de legajos > 0 –ciclo anidado-. Nota y grado, no están ordenados por grado. Por ejemplo, para el grado 1, ingresamos:</p> <table border="1" data-bbox="581 1350 1031 1560"> <thead> <tr> <th>Variable leg</th><th>variable grado</th><th>Variable nota</th></tr> </thead> <tbody> <tr> <td>123</td><td>A</td><td>8</td></tr> <tr> <td>345</td><td>B</td><td>10</td></tr> <tr> <td>103</td><td>A</td><td>7</td></tr> <tr> <td>839</td><td>A</td><td>4</td></tr> <tr> <td>...</td><td>...</td><td>...</td></tr> </tbody> </table> <ul style="list-style-type: none"> • <u>Cómo resolvemos los resultados parciales de 1er grado?</u> (el promedio de A y promedio de B), <u>si no sabemos cuántos legajos ingresarán?</u> Necesito sumar por separado las notas de A y las notas de B, pero además necesito contar esas notas para obtener el promedio porque desconocemos cuántos A y cuántos B existen. • Para ese proceso de sumar y contar podríamos tener una función, Proceso_2, auxiliar, que realice las operaciones. 	Variable leg	variable grado	Variable nota	123	A	8	345	B	10	103	A	7	839	A	4	<p>Esta función auxilia a Proceso_1, aquí recibimos nota y grado. Debemos sumar cada nota y contarla también. Es decir que por todos los legajos que se ingresen en cada grado esta función sumará y contará por A y B, recordando que vuelve a comenzar su proceso en el cambio de grado con contadores y acumuladores inicializados en 0.</p>
Variable leg	variable grado	Variable nota																		
123	A	8																		
345	B	10																		
103	A	7																		
839	A	4																		
...																		

	<p>Entonces le damos a esa función los valores, nota y grado, los pasaremos por valor porque no necesitamos modificar esas variables.</p> <ul style="list-style-type: none"> • <u>Pero hay que acumular y contar... y cómo vinculamos acumuladores y contadores con la Proceso_2?</u> De la misma manera que hicimos en main en la función Proceso_1, con pasajes de parámetros por dirección, pero con una salvedad, no usaremos una función de retorno porque, por cada año, sólo necesitamos sumar y contar por A y B. Cuando termine de procesar 1er grado, tendré en esos acumuladores y contadores lo que necesito para obtener los promedios. • Si lo resolvemos para 1er. grado, <u>sirve para el resto?</u> Claro que sí!, sólo que volvemos a empezar inicializando otra vez acumuladores y contadores para 2do grado, y debemos repetir esa inicialización para el resto. Es importante la inicialización de los acumuladores y contadores por cada grado, sino arrastrarían los valores de años anteriores procesados. 	
--	--	--