

Nota: Realice funciones recursivas solo en los ítems señalados.

Ejercicio 1

Generar un arreglo de registros que posea la siguiente información de 80 estudiantes: Nombre, Edad y Cantidad de materias rendidas.

Realizar un programa en C, que usando funcione/s para cada ítem permita:

- Cargar los datos de los estudiantes.
- Listar los datos de los estudiantes menores a una edad ingresada por teclado. (hacer una función recursiva)
- Indicar cuántos estudiantes tienen una cantidad de materias rendidas igual a 10 y cuantos tienen una cantidad menor. Realice una función que retorne un valor con la función y otro por parámetro; ambos valores mostrarlos en el main.

Ejercicio 2

Se tienen las notas de 25 estudiantes que cursan de 10 materias.

Generar una tabla y guardar esta información, donde las filas representen a las materias.

Realizar un programa en C, que usando funcione/s para cada ítem permita:

- Mostrar la calificación obtenida por cada estudiante en la materia 8. (genere una función recursiva que reciba la fila)
- Calcular la nota promedio de cada estudiante.

Ejercicio 3

Hacer el seguimiento de la ejecución y mostrar el estado de la memoria cuando se ejecuta la función Proceso.

Lote de prueba $n = 0$

```
int Proceso ( int n)
```

```
{  
    if ( n != 3 )  
        return n + Proceso ( n + 1 );  
    else  
        return (0);  
}
```

```
int main()
```

```
{  
    int num;  
    scanf("%d", &num);  
    printf("Resultado es %d", Proceso (num);  
}
```

Flores Medina

Parcial 1 Programación procedural

FECHA 7-09-22

Ejercicio N° 1

```
#include <stdio.h>
#define N 80
/* Algoritmo Ejercicio-1 */
```

```
struct estudiantes {
    char nombA[15];
    int edad, cantM;
} arr[N];
```

```
void carga(estudiantes arr[N]) {
    printf("A continuación se cargan los datos");
    for (int i=0; i<N; i++) {
        printf("Ingrese el nombre del alum.");
        fflush(stdin);
        gets(arr[i].nombA);
        printf("Ingrese la edad");
        scanf("%d", &arr[i].edad);
        printf("Ingrese cantidad de materias");
        scanf("%d", &arr[i].cantM);
    }
    return;
}
```

```
void listar(estudiantes arr[N], int ed, int i; int s) {
```

```
    if (i < 5) {
        if (arr[i].edad < ed) {
            printf("%s %d %d", arr[i].nombA, arr[i].edad, arr[i].cantM);
        }
        listar(arr, ed, i+1, s);
        else return;
    }
}
```

```
int contar(estudiantes arr[N], int &xc) {
```

```
    int c=0;
    for (int i=0; i<N; i++) {
        if (arr[i].cantM == 10) {
            c++;
        } else if (arr[i].cantM < 10) {
            xc++;
        }
    }
}
```

```
    return c;
}
```

Aprobado

/* Algoritmo principal */

```
int main() {
    int cuenta=0, edadRef, n=80;
```

```
    carga(arr);
    printf("Ingrese una edad de referencia");
    scanf("%d", &edadRef);
    listar(arr, edadRef, 0, n);
    printf("Estudiantes con cant M = 10");
    printf("%d", contar(arr, cuenta));
    printf("Estudiantes con cant M < 10");
    printf("%d", xc);
}
```


Ejercicio N° 2

```
#include <stdio.h>
#define N 25
#define M 10
void carga (int arr[M][N]) {
    for (int i=0; i<M; i++) {
        for (int j=0; j<N; j++) {
            printf("Ingrese la nota ");
            scanf("%d", &arr[i][j]);
        }
    }
    return;
}
```

```
void mostrar (int arr[M][N], int fila, int i, int s) {
    if (i == s) {
        return;
    } else {
        printf("%d", arr[fila-1][i]);
        mostrar(arr, fila, i+1, s);
    }
}
```

```
void promedio (int arr[M][N],
    for (int i=0; i<N; i++) {
        int sum=0;
        for (int j=0; j<M; j=j+N) {
            sum = sum + arr[i][j];
        }
        printf("Promedio %d", sum/M);
    }
    return;
}
```

```
/* Algoritmo Principal */
int main () {
    int arr[M][N], fila=8;
```

```
    carga(arr);
    printf("Calificaciones de cada estudiante para la materia %d", fila);
    mostrar(arr, fila, 0, 25);
    promedio(arr);
}
```

Florencia Molina

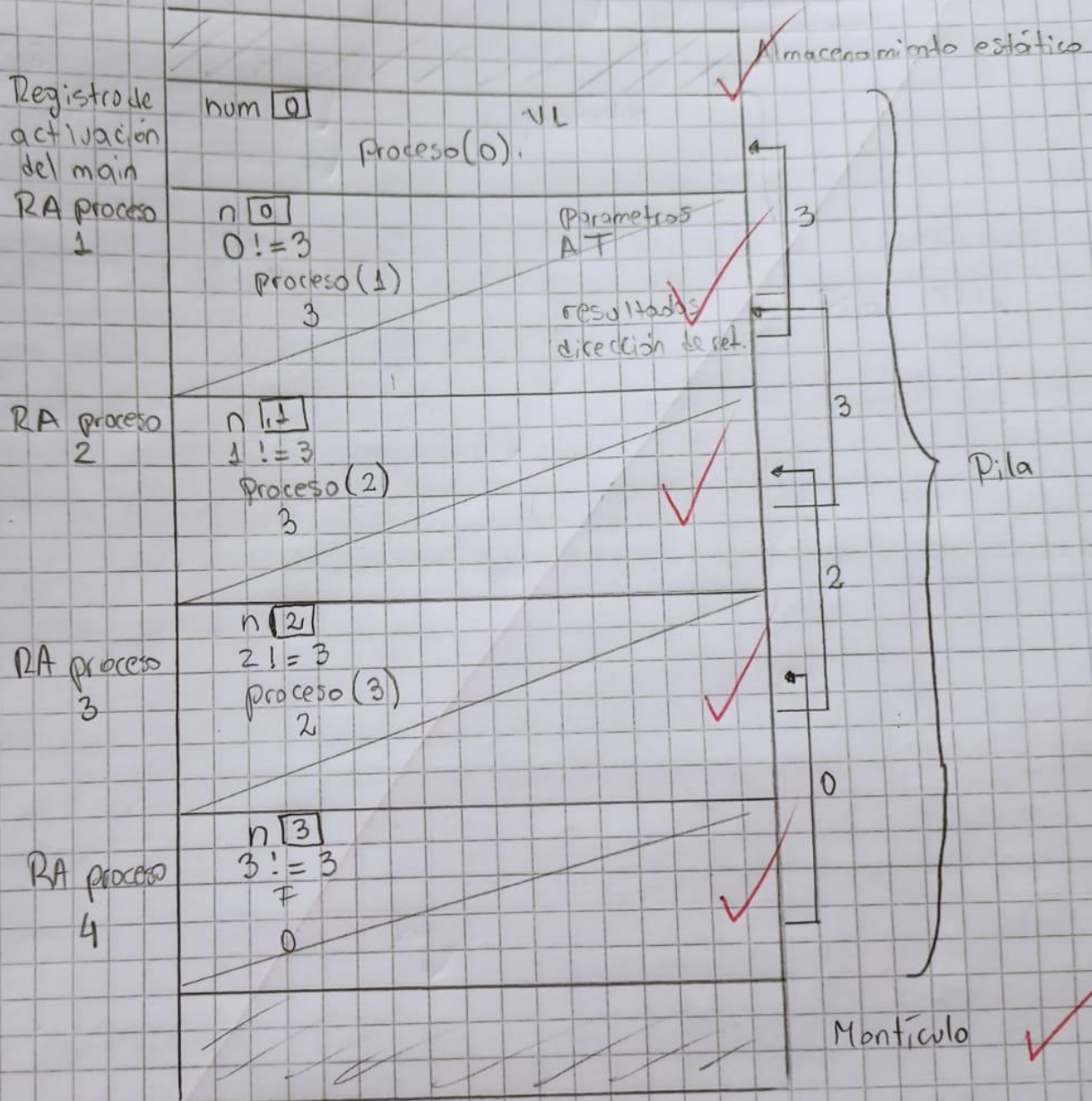
Ejercicio N° 3

LSI - 21.004

HOJA N° 2

FECHA 7-09-22

Mapa de memoria:



Salida	
Registro de activ. del main	Resultado es 3