



APRENDERAPROGRAMAR.COM

## PROGRAMACION EN LENGUAJE C: EJERCICIOS RESUELTOS CON ARRAYS O ARREGLOS Y CICLOS FOR Y WHILE (CU00540F)

Sección: Cursos

Categoría: Curso básico de programación en lenguaje C desde cero

Fecha revisión: 2031

**Resumen:** Entrega nº40 del curso básico "Programación C desde cero".

Autor: Mario Rodríguez Rancel

## EJERCICIO N°1: ENUNCIADO

Transformar en código el pseudocódigo que se muestra a continuación, relativo a un fichero con notas de 35 alumnos del que se quieren extraer datos estadísticos (% de aprobados, % de suspensos, número de alumnos con puntuación 10 y número de alumnos con puntuación 0). Considerar que los datos se encuentran en un fichero cuyo nombre es *notas.dat*. Razonar la lógica del programa.



```

1. Inicio [Algoritmo Notas de clase aprenderaprogramar.com]
2. Desde i = 1 hasta 35 Hacer
    2.1 Leer Nota(i)
    2.2 Si Nota(i) >= 5 Entonces
        Aprobados = Aprobados + 1
    SiNo
        Suspensos = Suspensos + 1
    FinSi
    2.3 Si Nota(i) = 10 Entonces
        Dieces = Dieces + 1
    FinSi
    2.4 Si Nota(i) = 0 Entonces
        Ceros = Ceros + 1
    FinSi
    Siguiente
3. Mostrar "% de aprobados es", (Aprobados / 35) * 100
4. Mostrar "% de suspensos es", (Suspensos / 35) * 100
5. Mostrar "Obtienen un 10 un total de", Dieces, "alumnos"
6. Mostrar "Obtienen un 0 un total de", Ceros, "alumnos"
7. Fin
  
```

## EJERCICIO N°1: SOLUCIÓN

Vamos a considerar un archivo que contenga los siguientes datos (nosotros usaremos un fichero donde habrá un dato en cada línea. Aquí los ponemos separados por guiones por motivos de espacio):

6 – 7 – 8 – 5 – 6 – 4 – 7 – 8 – 7 – 10 – 0 – 5 – 6 – 7 – 6 – 5 – 0 – 7 – 8 – 10 – 2 – 3 – 4 – 5 – 6 – 7 – 6 – 5 –  
4 – 10 – 4 – 3 – 2 – 3 – 4

El código sería el siguiente:

```
#include <stdio.h>
#include <stdlib.h>
#define NumeroAlumnos 35

int main() {
    double nota[NumeroAlumnos]; //35 elementos numerados de 0 a 34
    double aprobados;
    double suspensos;
    int dieces = 0;
    int ceros = 0;
    int i=0;

    puts ("Programa Notas de clase \n");
    //Extracción de datos de archivo
    FILE* fichero;
    fichero = fopen("c:\\notas.dat", "rt");
    for (i=0; i<=NumeroAlumnos-1; i++) {
        fscanf (fichero, "%lf", &nota[i] );
        if (nota[i] >= 5) { aprobados = aprobados + 1; }
        else { suspensos = suspensos + 1; }
        if (nota[i] == 10) { dieces = dieces + 1; }
        if (nota[i] == 0) { ceros = ceros + 1; }
    }
    fclose(fichero);

    printf ("Porcentaje de aprobados es %lf%% \n", (aprobados/NumeroAlumnos)*100);
    printf ("Porcentaje de suspensos es %lf%% \n", (suspensos/NumeroAlumnos)*100);
    printf ("Obtienen un 10 un total de %d alumnos \n", dieces);
    printf ("Obtienen un 0 un total de %d alumnos \n", ceros);

    return 0; // Ejercicios resueltos programacion C aprenderaprogramar.com
}
```

El resultado de ejecución del programa para los datos indicados como ejemplo sería:

#### Programa Notas

Porcentaje de aprobados es 65.714286 %

Porcentaje de suspensos es 34.285714 %

Obtienen un 10 un total de 3 alumnos

Obtienen un 0 un total de 2 alumnos

El bucle se puede plantear igualmente con un *do ... while (condición)*. Este sería el código del bucle (el resto del código se mantendría igual):

```

do {
    fscanf (fichero, "%lf", &nota[i] );
    printf ("Nota %d extraida: %lf\n", i, nota[i]);
    if (nota[i] >= 5) { aprobados = aprobados + 1; }
    else { suspensos = suspensos + 1; }
    if (nota[i] == 10) { dieces = dieces + 1; }
    if (nota[i] == 0) { ceros = ceros + 1; }
    i++; //Ejemplo aprenderaprogramar.com
} while (i<=34); // También válido i<35

```

## EJERCICIO N°2: ENUNCIADO

Se desea crear un programa que calcule la nota media de un grupo de 35 alumnos, a partir de los datos de notas contenidos en un fichero. Usar como archivo de datos el mismo que para el ejercicio anterior.

## EJERCICIO N°2: SOLUCIÓN

```

#include <stdio.h>
#include <stdlib.h>
#define NumeroAlumnos 35
// Ejercicios resueltos programacion C aprenderaprogramar.com
int main() {
    double nota[NumeroAlumnos]; //35 elementos numerados de 0 a 34
    double sumatorio = 0; double media = 0;
    int i=0;
    puts ("Programa Calcular Notas Medias \n");
    //Extracción de datos de archivo
    FILE* fichero;
    fichero = fopen("c:\notas.dat", "rt");
    for (i=0; i<=NumeroAlumnos-1; i++) {
        fscanf (fichero, "%lf", &nota[i] );
        sumatorio = sumatorio + nota[i];
    }
    fclose(fichero);
    media = sumatorio / NumeroAlumnos;
    printf ("La nota media de la clase es %lf\n", media);
    return 0;
}

```

El resultado de ejecución del programa para los datos de ejemplo sería el siguiente:

### Programa Calcular Notas Medias

```

La nota media de la clase es 5.428571
Process returned 0 (0x0) execution time : 0.046 s
Press any key to continue.

```

### EJERCICIO N°3: ENUNCIADO

Una central distribuye cal hacia diferentes almacenes sucursales. Disponen de un muelle de carga a donde van llegando sacas de cal de entre 3000 y 9000 Kgs, con pesos variables en función de las circunstancias de la producción. La empresa dispone de una flota de camiones con capacidades de carga de entre 18000 y 28000 Kgs.

Se pretende establecer un protocolo consistente en cargar 20 camiones diarios. Cada camión se quiere cargar como máximo a su límite de capacidad debiendo partir si con la siguiente saca en la línea de producción se fuera a exceder su capacidad.

La empresa quiere desarrollar un programa que le pida al operario encargado de carga la capacidad del camión y el peso de las sacas, indicándole si debe cargar la saca o despachar el camión para comenzar a cargar otro.

Nota: este ejercicio fue resuelto en pseudocódigo, donde además tenemos ejemplos numéricos que nos pueden permitir verificar que el programa funcione correctamente.

### EJERCICIO N°3: SOLUCIÓN

El código se incluye a continuación. Un posible resultado de ejecución es (los datos que hemos introducido son: Capacidad de camiones: número 1: 22000 kgs, número 2: 18000 kgs, número 3: 18000 kgs, número 4: 20000 kgs, número 5: 28000 kgs. Peso de las sacas: n°1: 3000 kgs, n°2: 5000 kgs, n°3: 9000 kgs, n°4: 3000 kgs, n°5: 3000 kgs, n°6: 8000 kgs, n°7: 9000 kgs, n°8: 7000 kgs, n°9: 7000 kgs, n°10: 9000 kgs, n°11: 4000 kgs, n°12: 5000 kgs (... continuaría hasta alcanzar el límite diario):

#### Programa distribucion de cal

Introduzca la capacidad del camion numero 1 en kgs: 22000  
Introduzca el peso de la saca numero 1 en kgs: 3000  
Cargue la saca numero 1.  
Introduzca el peso de la saca numero 2 en kgs: 5000  
Cargue la saca numero 2  
Introduzca el peso de la saca numero 3 en kgs: 9000  
Cargue la saca numero 3  
Introduzca el peso de la saca numero 4 en kgs: 3000  
Cargue la saca numero 4  
Introduzca el peso de la saca numero 5 en kgs: 3000  
No cargue todavía. Despache el camion numero 1 y ponga a cargar el camion numero 2  
Introduzca la capacidad del camion numero 2 en kgs: 18000  
Cargue la saca numero 5  
Introduzca el peso de la saca numero 6 en kgs: 8000  
Cargue la saca numero 6  
Introduzca el peso de la saca numero 7 en kgs: 9000  
No cargue todavía. Despache el camion numero 2 y ponga a cargar el camion numero 3  
Introduzca la capacidad del camion numero 3 en kgs: 18000  
Cargue la saca numero 7  
Introduzca el peso de la saca numero 8 en kgs: 7000  
Cargue la saca numero 8  
Introduzca el peso de la saca numero 9 en kgs: 7000  
No cargue todavía. Despache el camion numero 3 y ponga a cargar el camion numero 4

Introduzca la capacidad del camion numero 4 en kgs: 20000  
 Cargue la saca numero 9. Introduzca el peso de la saca numero 10 en kgs: 9000  
 Cargue la saca numero 10. Introduzca el peso de la saca numero 11 en kgs: 4000  
 Cargue la saca numero 11. Introduzca el peso de la saca numero 12 en kgs: 5000  
 No cargue todavia. Despache el camion numero 4 y ponga a cargar el camion numero 5  
 Introduzca la capacidad del camion numero 5 en kgs: 28000  
 Cargue la saca numero 12. Introduzca el peso de la saca numero 13 en kgs: ... (continuaría hasta terminar)

```

#include <stdio.h>
#include <stdlib.h>
// Ejercicios resueltos programacion C aprenderaprogramar.com
int main() {
    int i = 0; int j = 0; int capacidad = 0;
    int pesoSaca = 0; int cargado = 0;
    puts ("Programa distribucion de cal");
    for (i =1; i<=20; i++) {
        printf ("Introduzca la capacidad del camion numero %d en kgs: ", i);
        scanf ("%d", &capacidad);
        while (cargado + pesoSaca <= capacidad) {
            if (pesoSaca >0) {
                printf ("Cargue la saca numero %d \n", j);
                cargado = cargado + pesoSaca;
            }
            j = j+1;
            printf ("Introduzca el peso de la saca numero %d en kgs: ", j);
            scanf ("%d", &pesoSaca);
        } //Cierre del while
        if (i<20) {
            printf ("No cargue todavía. Despache el camion numero %d y ponga a cargar el camion numero %d \n", i,
i+1);
            cargado = 0;
        }
    } //Cierre del for
    printf ("No cargue esta saca. Despache el camion numero %d. FINALIZADA LA CARGA DEL DIA", i-1 );
    return 0;
}
  
```

#### EJERCICIO N°4: ENUNCIADO

Una ONG tiene puntos de reparto de vacunas que se pretende funcionen de la siguiente manera. Cada día, empezar con 1000 vacunas disponibles y a través de un programa que controla las entregas avisar si el inventario baja de 200 unidades. El pseudocódigo para el programa es el indicado a continuación. Razonar la lógica de este pseudocódigo y crear un programa a partir de él.

1. Inicio [Control de Vacunas ejemplo aprenderaprogramar.com]
2. Existencias = 1000
3. Mientras Existencias >= 200 Hacer
  - Mostrar "Introduzca el número de unidades entregadas"
  - Pedir Entregadas
  - Existencias = Existencias – Entregadas
- Repetir**
4. Mostrar "El inventario ha bajado de 200 unidades. Debe comunicarlo"
5. Fin

## EJERCICIO N°4: SOLUCIÓN

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    int existencias; int entregadas;
    puts ("Programa control de vacunas\n");
    existencias = 1000;
    while (existencias >=200) {
        printf ("Introduzca el numero de unidades entregadas: ");
        scanf ("%d", &entregadas);
        existencias = existencias - entregadas;
    } // Ejercicios resueltos programacion C aprenderaprogramar.com
    puts("\nAviso. El inventario ha bajado de 200 uds. Debe comunicarlo.");
    return 0;
}
```

Este podría ser un resultado de ejecución:

```
Programa control de vacunas
Introduzca el numero de unidades entregadas: 350
Introduzca el numero de unidades entregadas: 100
Introduzca el numero de unidades entregadas: 250
Introduzca el numero de unidades entregadas: 150
Aviso. El inventario ha bajado de 200 uds. Debe comunicarlo.
```

## EJERCICIO

Crea un programa que usando bucles nos permita pedir un número par comprendido entre 100 y 200 y nos muestre todos los números pares comprendidos entre el número facilitado y 200. Por ejemplo si el número facilitado es 192 nos debería mostrar 192, 194, 196, 198 y 200.

Para comprobar si tus respuestas son correctas puedes consultar en los foros [aprenderaprogramar.com](http://www.aprenderaprogramar.com).

**Próxima entrega:** CU00541F

**Acceso al curso completo** en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) --> Cursos, o en la dirección siguiente:

[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=82&Itemid=210](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=82&Itemid=210)