

# Trabajo Práctico N°1

Hugo Frey

Universidad Nacional de la Patagonia Austral

Paradigmas de Lenguajes de Programación - 2025

Dra. S. Casas y Mg. G. Vidal

# Trabajo Práctico N°1

## Gestión de calidad

### Actividades

1. Crear, tanto de forma recursiva como de forma iterativa, una función y diga si una cadena de caracteres es simétrica (un palíndromo). Por ejemplo, "DABALEARROZALAZORRAELABAD" es un palíndromo.
2. Hacer un programa que realice las funciones que a continuación se indican. Todas las opciones deben ser presentadas al usuario a través de un menú de opciones.
  - a. Escriba una función en la que se introduzcan 10 enteros y determine cuáles de estos enteros son pares y cuáles son impares.
  - b. Un número entero es un “número perfecto” si sus factores, incluyendo al 1 (pero excluyendo en el número mismo), suman igual que el número. Ejemplo: 6 es un numero perfecto porque  $6 = 1+2+3$ . Escriba una función que regrese los primeros 100 números perfectos. Esta función debe tener una función anidada que determine al número perfecto.
  - c. Escriba una función que tome un valor entero de cuatro dígitos y regrese el número con los dígitos invertidos. Por ejemplo, dado el número 7631, la función deberá regresar 1367.
3. Crear un programa que utilice una función que devuelva el mayor, el menor y la media de los valores de un array de números enteros. Luego crear otro programa para que la función devuelva punteros al mayor y el menor valor del array.

# Trabajo Práctico N°1

## Gestión de calidad

### Desarrollo

1.

```
int palindromoIterativo(char *cadena) {  
  
    int longitud = strlen(cadena);  
  
    int inicio = 0;  
  
    int fin = longitud - 1;  
  
    while (fin > inicio){  
  
        if (cadena[inicio] != cadena[fin]){  
  
            return 0;  
  
        }  
  
        inicio++;  
  
        fin--;  
  
    }  
  
    return 1;  
}  
  
int palindromoRecursivo(char *cadena, int inicio, int fin){  
  
    if (inicio > fin){  
  
        return 1;  
  
    }  
  
    if (cadena[inicio] != cadena[fin]){  
  
        return 0;  
  
    }else{
```

# Trabajo Práctico N°1

## Gestión de calidad

```
        return palindromoRecursoivo(cadena, inicio+1, fin-1);  
  
    }  
  
}
```

2.

```
#include <stdio.h>  
  
#include <stdlib.h>  
  
void numerosParesImpares () {  
  
    int numeros[10];  
  
    for (int i = 0; i < 10; i++){  
  
        printf("Ingrese un numero entero: ");  
  
        scanf("%d",numeros+i);  
  
    }  
  
    for (int i = 0; i < 10; i++){  
  
        if (numeros[i]%2 == 0){  
  
            printf("El número %d es par\n", numeros[i]);  
  
        }else{  
  
            printf("El número %d es impar\n", numeros[i]);  
  
        }  
  
    }  
  
}  
  
int esNumeroPerfecto(int numero) {
```

# Trabajo Práctico N°1

## Gestión de calidad

```
int sum = 0;

for (int i = 1; i < numero; i++){

    if (numero % i == 0) {

        sum += i;

    }

}

return sum == numero;
}

void primero100NumerosPerfectos() {

    int contador = 0;

    int num = 2;

    while (contador<100){

        int condicion = esNumeroPerfecto(num);

        if (condicion){

            contador++;

            printf("%d es un numero perfecto\n", num);

        }

        num++;

    }

}

void inversoNumero() {
```

# Trabajo Práctico N°1

## Gestión de calidad

```
int numero = 0;

int divisor = 10;

printf("Ingrese un numero de 4 digitos: ");

scanf("%d", &numero);

while (numero < 1000 || numero >= 10000){

    printf("Error, porfavor ingrese un numero de 4 digitos: ");

    scanf("%d", &numero);

}

printf("El inverso del numero es: ");

int modulo;

while (numero > 0){

    modulo = numero % divisor;

    printf("%d", modulo);

    numero -= modulo;

    numero /= divisor;

}

printf("\n");

}

int main(){

    int opcion=-1;

    while (opcion != 0){

        printf("Bienvenido al menu de opciones\n");

        printf("1. Números pares e impares\n");
```

# Trabajo Práctico N°1

## Gestión de calidad

```
printf("2. 100 números perfectos\n");

printf("3. Inverso\n");

printf("0. Salir\n");

printf("Que opcion quiere elegir: ");

scanf("%d", &opcion);

switch (opcion){

case 1:

    numerosParesImpares();

    break;

case 2:

    primero100NumerosPerfectos();

    break;

case 3:

    inversoNumero();

    break;

case 0:

    break;

default:

    break;

}

}

return 0;

}
```

# Trabajo Práctico N°1

## Gestión de calidad

3.

```
#include <stdio.h>

#include <stdlib.h>

float media (int *numeros, int tamaño){

    int suma = 0;

    for (int i = 0; i < tamaño; i++){

        suma += numeros[i];

    }

    return (float) suma / tamaño;

}

int mayorValor(int *numeros, int tamaño){

    int mayorValor = numeros[0];

    for (int i = 1; i < tamaño; i++){

        if (mayorValor < numeros[i]){

            mayorValor = numeros[i];

        }

    }

    return mayorValor;

}

int menorValor(int *numeros, int tamaño){

    int menorValor = numeros[0];
```



# Trabajo Práctico N°1

## Gestión de calidad

```
for (int i = 1; i < tamano; i++){

    if (menorValor > numeros[i]){

        menorValor = numeros[i];

    }

}

return menorValor;

}

int* mayorValorPuntero(int *numeros, int tamano){

    int *mayorValor = &numeros[0];

    for (int i = 1; i < tamano; i++){

        if (*mayorValor < numeros[i]){

            mayorValor = &numeros[i];

        }

    }

    return mayorValor;

}

int* menorValorPuntero(int *numeros, int tamano){

    int *menorValor = &numeros[0];

    for (int i = 1; i < tamano; i++){

        if (*menorValor > numeros[i]){

            menorValor = &numeros[i];

        }

    }

}
```

# Trabajo Práctico N°1

## Gestión de calidad

```
}

return menorValor;

}

void valoresEstadisticos(int *numeros, float *valores, int tamano){

    valores[0] = menorValor(numeros, tamano);

    valores[1] = mayorValor(numeros, tamano);

    valores[2] = media(numeros, tamano);

}

int main(){

    int numeros[] = {6,7,6,8,8,9,6,7,10,10};

    float valores[3];

    int tamano = sizeof(numeros) / sizeof(numeros[0]);

    valoresEstadisticos(numeros, valores, tamano);

    printf("El valor minimo es %f, El valor maximo es %f y la media es %f\n", valores[0], valores[1], valores[2]);

    printf("La direccion del valor minimo es %p y el valor maximo es %p", (void *) menorValorPuntero(numeros, tamano), (void *) mayorValorPuntero(numeros, tamano));

}
```