

Programación de Computadores

2023-2

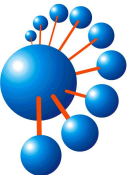
Tema 3: Ciclos y condiciones



Universidad
de Concepción

José Fuentes - jfuentess@inf.udec.cl

Departamento de
Ingeniería Informática y
Ciencias de la Computación



Estructura selectiva if-else

— — —

```
if(condición A) {  
    ... código ...  
}
```

```
if(condición A) {  
    ... código ...  
} else {  
    ... código ...  
}
```

```
float PRO;  
printf("ingrese el promedio del alumno: ");  
scanf("%f", &PRO);  
if (PRO >= 6)  
    printf("\nAprobado");
```

```
float PRO;  
printf("Ingrese el promedio del alumno: ");  
scanf("%f", &PRO);  
if (PRO >= 6.0)  
    printf("\nAprobado");  
else  
    printf("\nReprobado");
```

Estructura selectiva if-else

— — —

```
if(condición A) {  
    ... código ...  
} else if(condición B) {  
    ... código ...  
} else if(condición C) {  
    ... código ...  
} else {  
    ... código ...  
}
```

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

```
/* ventas descendentes.
```

```
El programa, al recibir como datos tres valores que representan las ventas  
de los vendedores de una tienda de discos, escribe las ventas en  
orden descendente.
```

```
P, S y R: variables de tipo real.    */
```

```
void main(void)
```

```
{
```

```
float P, S, R;
```

```
printf("\nIngrese las ventas de los tres vendedores: ");
```

```
scanf("%f %f %f", &P, &S, &R);
```

```
if (P > S)
```

```
    if (P > R)
```

```
        if (S > R)
```

```
            printf("\n\n El orden es P, S y R: %8.2f %8.2f %8.2f", P, S, R);
```

```
        else
```

```
            printf("\n\n El orden es P, R y S: %8.2f %8.2f %8.2f", P, R, S);
```

```
    else
```

```
        printf("\n\n El orden es R, P y S: %8.2f %8.2f %8.2f", R, P, S);
```

```
else
```

```
    if (S > R)
```

```
        if (P > R)
```

```
            printf("\n\n El orden es S, P y R: %8.2f %8.2f %8.2f", S, P, R);
```

```
        else
```

```
            printf("\n\n El orden es S, R y P: %8.2f %8.2f %8.2f", S, R, P);
```

```
    else
```

```
        printf("\n\n El orden es R, S y P: %8.2f %8.2f %8.2f", R, S, P);
```

```
}
```

Estructura selectiva switch-case

```
switch(selector) {  
    case valor1:  
        ... código ...  
        break;  
    case valor2:  
        ... código ...  
        break;  
    ...  
    case valorN:  
        ... código ...  
        break;  
    default:  
        ... código ...  
        break;  
}
```

```
#include <stdio.h>  
#include <math.h>  
  
/* Función matemática.  
El programa obtiene el resultado de una función.  
  
OP y T: variables de tipo entero.  
RES: variable de tipo real. */  
  
void main(void)  
{  
    int OP, T;  
    float RES;  
    printf("Ingrese la opción del cálculo y el valor entero: ");  
    scanf("%d %d", &OP, &T);  
    switch(OP)  
    {  
        case 1: RES = T / 5;  
            break;  
        case 2: RES = pow(T,T);  
            /* La función pow está definida en la biblioteca math.h */  
            break;  
        case 3:  
        case 4: RES = 6 * T/2;  
            break;  
        default: RES = 1;  
            break;  
    }  
    printf("\nResultado:   %7.2f", RES);  
}
```

Estructura iterativa for

— — —

```
for(valor inicial; condición término; incremento/decremento) {  
    ... código ...  
}
```

```
#include <stdio.h>  
/* Nómina.  
El programa, al recibir los salarios de 15 profesores,  
obtiene el total de la nómina de la universidad. */  
void main(void) {  
    int I;  
    float SAL, NOM;  
    NOM = 0;  
    for (I=1; I<=15; I++) {  
  
        printf("\Ingrese el salario del profesor%d:\t", I);  
        scanf("%f", &SAL);  
        NOM = NOM + SAL;  
    }  
    printf("\nEl total de la nómina es: %.2f", NOM);  
}
```

Estructura iterativa while

— — —

```
while(condición para continuar) {  
    ... código ...  
}
```

```
#include <stdio.h>  
#include <math.h>  
/* Suma cuadrados.  
El programa, al recibir como datos un grupo de enteros  
positivos, obtiene el cuadrado de los mismos y la suma  
correspondiente a dichos cuadrados. */  
void main(void) {  
    int NUM;  
    long CUA, SUC = 0;  
    printf("\nIngrese un número entero -0 para terminar-:\t");  
    scanf("%d", &NUM);  
  
    /* Observa que la condición es verdadera mientras el  
       entero es diferente de cero. */  
    while (NUM) {  
        CUA = pow (NUM, 2);  
        printf("%d al cubo es %ld", NUM, CUA);  
        SUC = SUC + CUA;  
        printf("\nIngrese un número entero -0 para terminar-:\t");  
        scanf("%d", &NUM);  
    }  
    printf("\nLa suma de los cuadrados es %ld", SUC);  
}
```

Estructura iterativa do-while

— — —

```
do {  
    ... código ...  
} while(condición para continuar)
```

```
#include <stdio.h>  
/* Suma pagos.  
El programa obtiene la suma de los pagos realizados el  
último mes.*/  
  
void main(void) {  
    float PAG, SPA = 0;  
    printf("Ingrese el primer pago:\t");  
    scanf("%f", &PAG);  
    /* Observa que al utilizar la estructura do-while al  
    menos se necesita un pago.*/  
    do {  
        SPA = SPA + PAG;  
        printf("Ingrese el siguiente pago -0 para terminar-:\t ");  
        scanf("%f", &PAG);  
    }  
    while (PAG);  
  
    printf("\nEl total de pagos del mes es: %.2f", SPA);  
}
```

if-else VS switch-case

```
switch(sel) {  
  case valor1:  
    ... código ...  
    break;  
  case valor2:  
    ... código ...  
    break;  
  ...  
  case valorN:  
    ... código ...  
    break;  
  default:  
    ... código ...  
    break;  
}
```



```
if(sel == valor1) {  
  ... código ...  
}  
else if(sel == valor2) {  
  ... código ...  
}  
...  
else if(sel == valorN) {  
  ... código ...  
}  
else {  
  ... código ...  
}
```


for VS while

— — —

```
for(exp1;exp2;exp3) {  
    ... código ...  
}
```



```
exp1;  
while(exp2) {  
    ... código ...  
    exp3;  
}
```

ciclos infinitos

```
for(;;) {  
    ... código ...  
}
```

```
while(1) {  
    ... código ...  
}
```

continue y break

— — —

continue

```
#include <stdio.h>

int main() {
    int i;
    double number, sum = 0.0;

    for (i = 1; i <= 10; ++i) {
        printf("Enter a n%d: ", i);
        scanf("%lf", &number);

        // Si se ingresa un número negativo,
        // omitirlo sin terminar el ciclo
        if (number < 0.0) {
            continue;
        }

        sum += number; // sum = sum + number;
    }

    printf("Sum = %.2lf", sum);

    return 0;
}
```

break

```
#include <stdio.h>

int main() {
    int i;
    double number, sum = 0.0;

    for (i = 1; i <= 10; ++i) {
        printf("Enter a n%d: ", i);
        scanf("%lf", &number);

        // Si se ingresa un número negativo,
        // terminar el ciclo
        if (number < 0.0) {
            break;
        }

        sum += number; // sum = sum + number;
    }

    printf("Sum = %.2lf", sum);

    return 0;
}
```

Contexto de validez de una variable

— — —

```
#include <stdio.h>

void main() {
    int i = 9, j = 0;

    if(j == 0) {
        int i = 10;

        printf("(1) i: %d\n", i);
    }
    printf("(2) i: %d\n", i);
}
```

```
#include <stdio.h>

void main() {
    int i = 9, j = 5;

    for(int i=1; i <= j; i++) {
        printf("i: %d\n", i);
    }
    printf("[*] i: %d\n", i);
}
```

¿Qué imprime cada programa?

¡A practicar!

— — —

Ejemplo 1:
`probabilidad_rand.c`

Ejemplo 2:
`piramide.c`

Ejemplo 3:
`condicional_par_impar.c`

Ejemplo 4:
`menu_desplazamiento.c`

Ejemplo 5:
`estimacion_pi.c`

