

```
Funciones() {  
    char por[] = "Daniel Eduardo Castañeda Lucero"  
                "Eduardo Santiago Torres Coria";  
    char clase[] = "Fundamentos de Programación";  
    char grupo[] = "1CV35";  
}
```

entrada → *salida*

```
funcion() {  
    if(condición) {  
        // Bloque de código 1  
    } else {  
        // Bloque de código 2  
    }  
}
```

ingredientes → *resultado*

```
main() {  
    float numero;  
    printf("Ingrese un numero: ");  
    scanf("%f", &numero);  
    getch();  
}
```

```
tipo-de-dato nombreDeLaFuncion(parametros) {  
    int variables, sentenciaDeRetorno;  
    printf("Sentencias de ejecución.");  
    scanf("%d", &variables);  
    return sentenciaDeRetorno;  
}
```

```
tipo-de-dato nombreDeLaFuncion(parametros) {  
    // Bloque de código  
}
```

```
tipo-de-dato nombreDeLaFuncion(void) {  
    // Bloque de código  
}
```



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

tipo-de-dato funcion(void);

main(void) {
    // Bloque de código
}

tipo-de-dato funcion(void) {
    // Bloque de código
}
```

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

```
#include <conio.h>
```

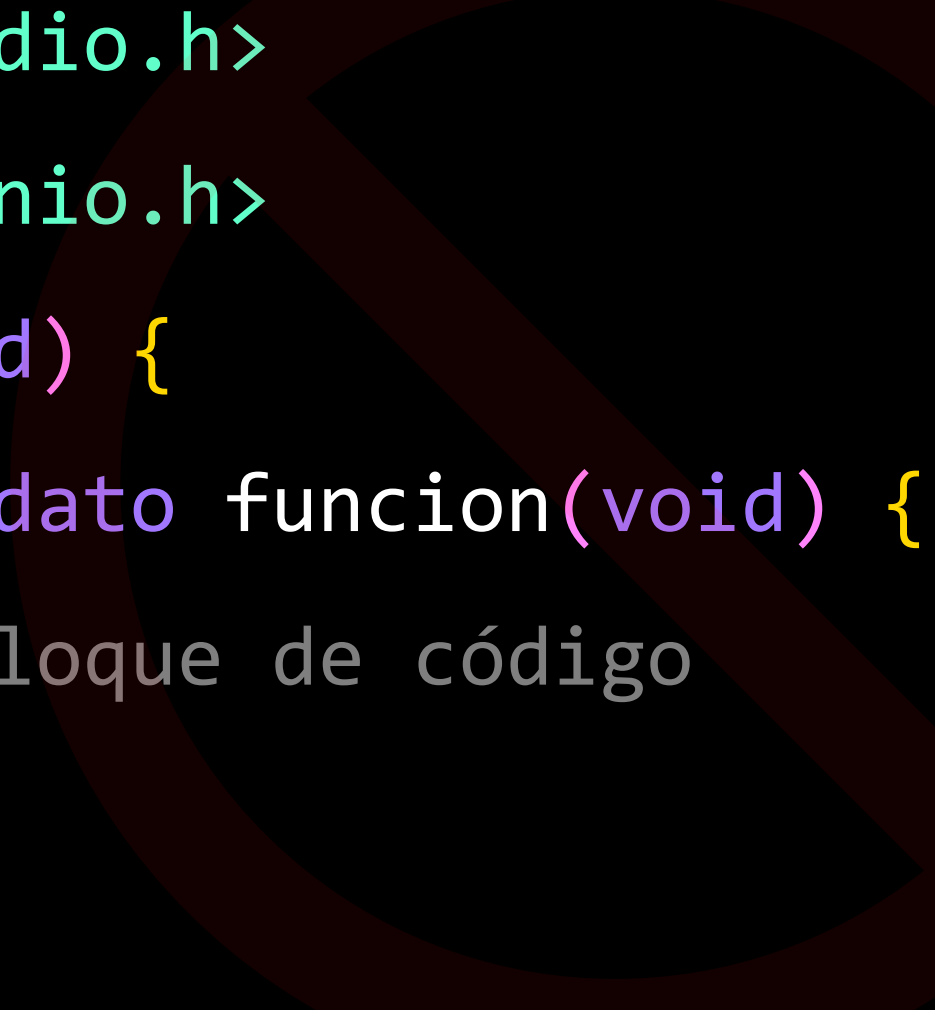
```
int main(void) {
```

```
    // Bloque de código
```

```
    return 0;
```

```
}
```

```
#include <stdio.h>
#include <conio.h>
int main(void) {
    tipo-de-dato funcion(void) {
        // Bloque de código
    }
}
```



```
#include <stdio.h>
#include <conio.h>
static tipo-de-dato funcion1(void);
extern tipo-de-dato funcion2(void);
int main(void) {
    // Bloque de código
}
```

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

static tipo-de-dato funcion(void);

int main(void) {
    // Bloque de código
}

tipo-de-dato funcion(void) {
    // Bloque de código
}
```

```
#include <stdio.h>
#include <conio.h>
extern tipo-de-dato funcion(void);
int main(void) {
    // Bloque de código
}
tipo-de-dato funcion(void) {
    // Bloque de código
}
```

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

tipo-de-dato funcion(void);

int main(void) {
    // Bloque de código
}

tipo-de-dato funcion(void) {
    // Bloque de código
}
```

```
tipo-de-dato tiposDeFunciones(void) {  
    // Bloque de código  
}
```



```
void funcionesSinRetorno(void) {  
    // Bloque de código  
}
```

```
float funcionesConRetorno(void) {  
    float variableDeRetorno;  
    // Bloque de código  
    return variableDeRetorno;  
}
```

```
int factorial(int numero) {  
    if(numero <= 1) {  
        return 1;  
    } else {  
        return numero * factorial(numero - 1);  
    }  
}
```

```
#include <stdio.h>
#include <conio.h>
void funcion1(void);
float funcion2(void);
int main(void) {
    // Bloque de código
}
```

```
int main(void) {  
    funcion();  
    return 0;  
}  
  
void funcion(void) {  
    // Bloque de código  
}
```

```
int main(void) {  
    float variable;  
    variable = funcion();  
    return 0;  
}  
  
float funcion(void) {  
    return 1.0;  
}
```

```
int main(void) {  
    printf("El retorno es: %f", funcion());  
    return 0;  
}  
  
float funcion(void) {  
    return 1.0;  
}
```

float variables;


```
#include <stdio.h>
#include <conio.h>
float variablesGlobales;
int main(void) {
    // Bloque de código
}
```

```
#include <stdio.h>
#include <conio.h>
static float variableGlobal1;
extern float variableGlobal2;
int main(void) {
    // Bloque de código
}
```

```
#include <stdio.h>
#include <conio.h>
int main(void) {
    float variablesLocales;
    // Bloque de código
}
```

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

```
#include <conio.h>
```

```
int main(void) {
```

```
    for(int i = 0; i < 10; i++) {
```

```
        float variableLocal1;
```

```
    }
```

```
}
```

```
#include <stdio.h>
#include <conio.h>
int main(void) {
    static float variableLocal1;
    automatic float variableLocal2;
    // Bloque de código
}
```

```
int main(void) {  
    incremento();  
    incremento();  
    return 0;  
}  
  
void incremento(void) {  
    static float variableEstatica = 0;  
    variableEstatica++;  
    printf("Variable: %f", variableEstatica);  
}
```

```
int main(void) {  
    incremento();  
    incremento();  
    return 0;  
}  
  
void incremento(void) {  
    automatic float variableEstatica = 0;  
    variableEstatica++;  
    printf("Variable: %f", variableEstatica);  
}
```

```
int main(void) {  
    funcion(argumento);  
    return 0;  
}
```

```
void funcion(int parametro) {  
    // Bloque de código  
}
```



```
int main(void) {  
    int argumento = 5;  
    incremento(argumento);  
    printf("Valor del argumento: %d\n", argumento);  
    return 0;  
}  
  
void incremento(int parametro) {  
    parametro++;  
    printf("Valor del parametro: %d\n", parametro);  
}
```



♥ 🏠 ~ ./program

Valor del parametro: 6

Valor del argumento: 5

♥ 🏠 ~

```
int main(void) {  
    funcion(&argumento);  
    return 0;  
}  
  
void funcion(int *parametro) {  
    // Bloque de código  
}
```

```
int main(void) {  
    int argumento = 5;  
    incremento(&argumento);  
    printf("Valor del argumento: %d\n", argumento);  
    return 0;  
}  
  
void incremento(int *parametro) {  
    (*parametro)++;  
    printf("Valor del parametro: %d\n", parametro);  
}
```



♥ 🏠 ~ ./program

Valor del parametro: 6

Valor del argumento: 6

♥ 🏠 ~

```
int main(void) {  
    funcion(argumento);  
    return 0;  
}
```

```
void funcion(const int parametro) {  
    // Bloque de código  
}
```

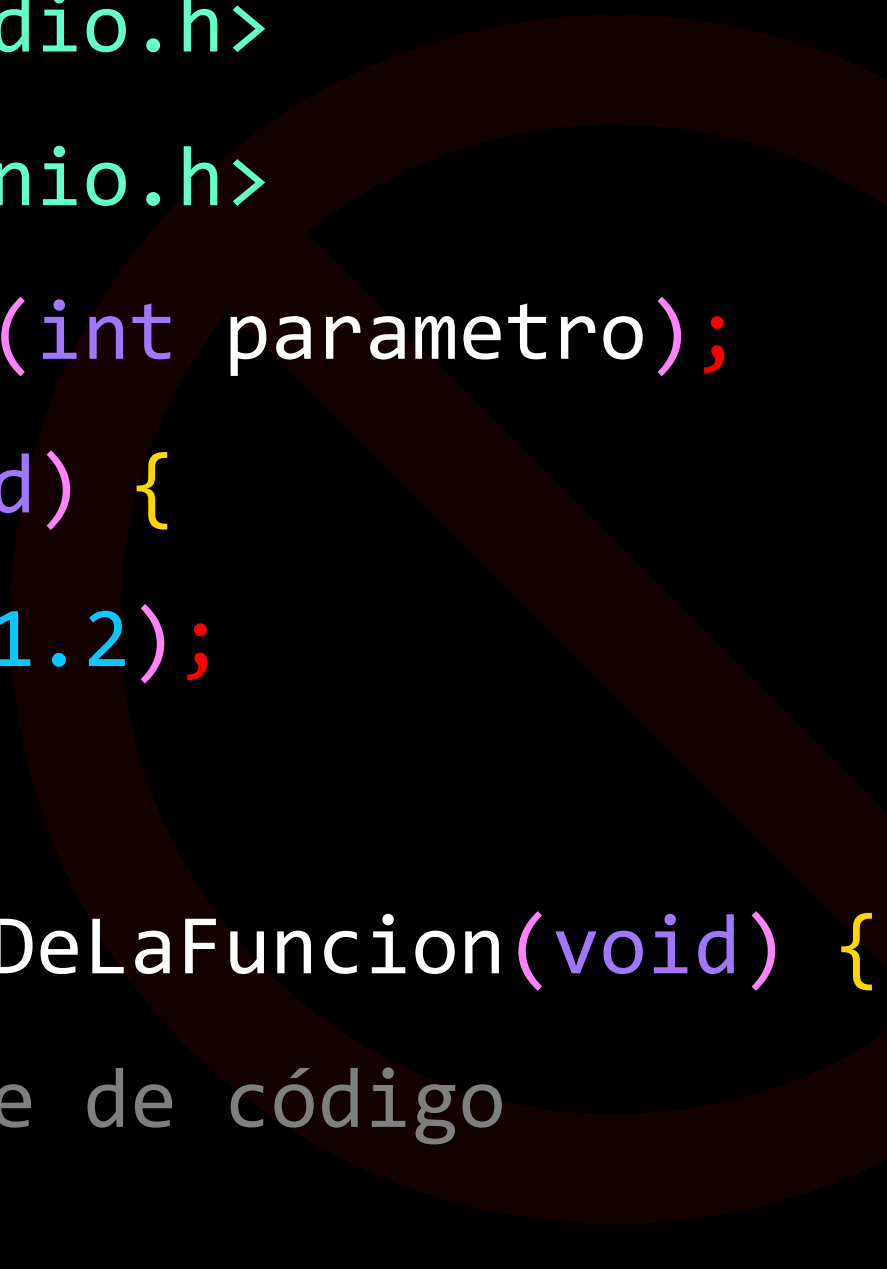
```
void conclusion(void) {  
    // Bloque de código  
}
```

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

void funcion(int parametro);

int main(void) {
    funcion(1.2);
}

float nombreDeLaFuncion(void) {
    // Bloque de código
}
```




```
#include <stdio.h>
#include <conio.h>
void funcion(void);
int main(void) {
    // Bloque de código
}
void funcion(void) {
    // Bloque de código
}
```