

L Hafidl Alkhair
TRKJ 1C
2023903430060

Struktur
&Fungsi

Struktur

- Struktur dalam Bahasa C adalah cara untuk mengorganisasi data dengan menggabungkan tipe data berbeda ke dalam satu entitas.
- Penjelasan tentang elemen-elemen yang dapat dimasukkan ke dalam struktur, seperti variabel dengan tipe data berbeda, yang memungkinkan penyusunan data yang lebih kompleks
- lustrasi contoh struktur data, misalnya penggunaan struktur untuk merepresentasikan informasi kontak dengan nama, nomor telepon, dan alamat email.



Fungsi

- Fungsi dalam Bahasa C adalah blok kode yang dapat dipanggil untuk menjalankan tugas tertentu. Fungsi membantu memecah program menjadi bagian-bagian yang lebih kecil dan terorganisir.
- Penjelasan tentang bagaimana fungsi dapat digunakan untuk mengeksekusi operasi matematika, operasi logika, atau tugas-tugas khusus lainnya.
- Pengertian parameter (input) yang diterima oleh fungsi dan nilai yang dikembalikan (output) oleh fungsi setelah operasi selesai dieksekusi.
- Ilustrasi contoh fungsi sederhana, seperti fungsi untuk menghitung luas lingkaran, dengan penjelasan langkah demi langkah.



Program fungsi:Fibonacci

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

```
int fibonacci(int n) {  
    if (n <= 1)  
        return n;  
    return fibonacci(n - 1) + fibonacci(n - 2);  
}
```

```
int main() {  
    int n = 10; // Ubah nilai n sesuai kebutuhan  
    printf("Deret Fibonacci hingga suku ke-%d:  
\\n", n);  
    for (int i = 0; i < n; i++) {  
        printf("%d ", fibonacci(i));  
    }  
    return 0;  
}
```

Cara Kerja Program:

Fungsi Fibonacci:

- Fungsi `fibonacci(n)` mengambil suku ke- n sebagai parameter.
- Jika n kurang dari atau sama dengan 1, fungsi mengembalikan nilai n (basis kasus).
- Jika n lebih besar dari 1, fungsi memanggil dirinya sendiri dua kali, yaitu untuk `n - 1` dan `n - 2`, dan mengembalikan hasil penjumlahan dari dua panggilan rekursif ini.

Panggilan Rekursif:

- Setiap kali fungsi `fibonacci(n)` dipanggil, itu akan memanggil dirinya sendiri untuk dua nilai yang lebih kecil.
- Ini terus berlanjut hingga mencapai basis kasus ($n \leq 1$), dan dari situ, nilai-nilai kembali dihitung secara rekursif.

Iterasi dalam `main()`:

- Dalam fungsi `main()`, program mengiterasi dari 0 hingga `n - 1`.
- Pada setiap iterasi, nilai deret Fibonacci untuk suku ke- i dihitung dan dicetak menggunakan fungsi `fibonacci(i)`.

Hasil Keluaran Program (Misalnya, jika `n = 10`):

` ``

Deret Fibonacci hingga suku ke-10:

0 1 1 2 3 5 8 13 21 34

` ``

Penjelasan hasil keluaran:

- `fibonacci(0)` = 0
- `fibonacci(1)` = 1
- `fibonacci(2)` = 1 (0 + 1)
- `fibonacci(3)` = 2 (1 + 1)
- `fibonacci(4)` = 3 (1 + 2)
- `fibonacci(5)` = 5 (2 + 3)
- `fibonacci(6)` = 8 (3 + 5)
- `fibonacci(7)` = 13 (5 + 8)
- `fibonacci(8)` = 21 (8 + 13)
- `fibonacci(9)` = 34 (13 + 21)

Program ini mencetak deret Fibonacci hingga suku ke-10 (sesuai nilai `n`) sebagai hasil keluarannya. Semakin besar nilai `n`, semakin panjang deret Fibonacci yang dihasilkan oleh program.

KESIMPULAN

Pemahaman tentang struktur dan fungsi dalam Bahasa C sangat penting karena:

1. ****Organisasi Data Efisien:****

- Struktur membantu mengelompokkan data terkait untuk pengelolaan yang efisien.

2. ****Modularitas Kode:****

- Fungsi memungkinkan pemecahan masalah menjadi bagian-bagian kecil yang dikelola secara terpisah, meningkatkan keterbacaan dan pemeliharaan kode.

3. ****Reusabilitas dan Optimalisasi:****

- Struktur dan fungsi dapat digunakan kembali, menghemat waktu pengembangan, dan memungkinkan penulisan kode yang efisien dan kreatif.

4. ****Manajemen Sumber Daya:****

- Memahami struktur dan fungsi membantu dalam manajemen memori dan sumber daya komputer, mencegah kebocoran memori dan kerugian sumber daya.

Dengan pemahaman yang baik tentang struktur dan fungsi, pengembang dapat membuat program yang efisien, mudah dimengerti, dan mudah dipelihara.