Test#1

Name : Joseph Janone Tiwouw

Data Structure

1. Jelaskan pemahaman anda tentang Struktur Data !
2. Jelaskan pemahaman anda tentang :
   1. Graph
   2. Linked List
   3. Array

Notes : Anda dapat menggambarkan struktur dari ke-3 struktur data dan memberikan penjelasan singkat.

1. Jelaskan mengapa ada banyak macam struktur data ? Mengapa tidak 1 saja struktur data yang ada ?
2. Berikan 1 contoh aplikasi dari Struktur data dan struktur data apa yang digunakan dan bagaimana struktur data itu digunakan !
3. Jelaskan bagaimana caranya kita mengukur performansi dari suatu algoritma ! Mengapa kita perlu menganlisa algoritma ?
4. Jelaskan pemahaman anda tentang Time Complexity !
5. Sebutkan dan jelaskan jenis-jenis time complexity yang ada.
6. Apa yang menjadi Big O Notation dari ekspresi berikut :
   1. 10 n + 7n + 18
   2. +
7. Urutkanlah ekspresi yang ada di no. 8 dari yang terbesar sampai yang terkecil
8. Jika ada 4 algoritma dengan ekspresi seperti pada no. 8, algoritma dengan ekspresi yang mana yang akan anda pilih ? Mengapa ?
9. Buatlah algoritma untuk menampilkan pola \* seperti berikut ini :

\*

\* \*

\* \* \*

Tuliskanlah Big O Notation yang menggambarkan time complexity dari algoritma yang sudah anda buat !

1. Sebuah struktur data adalah cara penyimpanan, penyusunan dan pengaturan data di dalam media penyimpanan komputer sehingga data tersebut dapat digunakan secara efisien dan merupakan sesuatu yang berfungsi untuk mengorganisasikan data sedemikian rupa sehingga implementasi ( penerapan ) atau pemeliharaan logika program menjadi lebih terstruktur.
2. A. Graph merupakan struktur data yang non-linear dan terdiri dari kumpulan node. Node-node tersebut disambungkan oleh suatu garis tertentu. Biasanya digambarkan sebagai kumpulan titik-titik (melambangkan verteks) yang dihubungkan oleh garis-garis (melambangkan sisi).

B. Linked List adalah daftar yang terhubung berupa potongan-potongan aneka macam data yang tersimpan secara acak. Linked list digambarkan Setiap node memiliki data dan referensi ke node selanjutnya dan atau ke node sebelumnya.

C. Array adalah struktur data yang memiliki kumpulan elemen dan elemen-elemen tersebut jaraknya saling berdekatan dalam memori. Digambarkan Setiap elemen diakses langsung melalui indeksnya. Indeks array harus berupa tipe data yang menyatakan urutan, misalnya: integer atau karakter.

1. Karena masing-masing jenis struktur data yang berbeda cocok untuk penggunaan-penggunaan yang berbeda, dan beberapa jenis memang dikhususkan untuk tugas tertentu.
2. #include <stdio.h>

void main(){

// isi awal array

char huruf[5] = {'a', 'b', 'c', 'd', 'e'};

// mengubah isi array

huruf[2] = 'z';

// mencetak isi array

printf("Huruf: %c\n", huruf[2]);

}

Aplikasi untuk mengisi ulang data pada array, menggunaka n struktur data array

1. Alasannya adalah Algoritma membantu kita memahami skabilitas sebuah program. Performa algoritma terkadang menjadi pembeda antara yang mungkin dilakukan dan yang tidak mungkin dilakukan. Analisis algoritma memberi gambaran informasi tentang perilaku sebuah program.
2. Time Complexity adalah jumlah waktu yang dibutuhkan oleh suatu algoritma untuk berjalan, sebagai fungsi dari panjang input. Ini mengukur waktu yang dibutuhkan untuk mengeksekusi setiap pernyataan kode dalam suatu algoritma.
3. **A. Constant Time** artinya banyaknya input yang diberikan kepada sebuah algoritma, tidak akan mempengaruhi waktu proses (runtime) dari algoritma tersebut.

**B. Logarithmic Time** artinya ketika kita memberikan input sebesar n terhadap sebuah fungsi, jumlah tahapan yang dilakukan oleh fungsi tersebut berkurang berdasarkan suatu faktor. Salah satu contohnya adalah algoritma **Binary Search.**

**C. Linear Time**adalah ketika runtime dari fungsi kita berbanding lurus dengan jumlah input yang diberikan.

**D. Quadratic Time** adalah ketika runtime dari fungsi kita adalah sebesar n^2, dimana n adalah jumlah input dari fungsi tersebut. Hal tersebut bisa terjadi karena kita menjalankan **fungsi linear didalam fungsi linear**(n\*n).

**E. Exponential Time** biasanya digunakan dalam situasi dimana kita tidak terlalu tahu terhadap permasalahan yang dihadapi, sehingga mengharuskan kita mencoba setiap kombinasi dan permutasidari semua kemungkinan.

* 1. 10 n + 7n + 18 = -> O (n^2) quadratical
  2. + = -> O (n^10) polynomial
  3. = -> O (log(n)) linear logaritmic
  4. = -> O (n^25) polynomial

10 n + 7n + 18 = -> O (n^2)

+ = -> O (n^10)

= -> O (n^25)

= -> O (log(n))

1. 10 n + 7n + 18 = -> O (n^2) quadratical, karena lebih kecil dan lebih mudah di olah
2. 4.

#include <iostream>

using namespace std;

int main()

{

int i, j, n;

cout << "Masukkan jumlah baris: ";

cin >> n;

for (i = 1; i <= n; i++)

{

for (j = 1; j <= i; j++)

{

cout << "\* ";

}

//Bagian akhir

cout << "\n";

}

return 0;

}