**Ujian Saringan Masuk Suitmedia**

**Bobot Soal : 20**

**Nama Kandidat : Fardan Nozami Ajitama**

**Posisi Yang Dipilih : Backend Engineer**

Mohon implementasikan fungsi *hitungNomorBit()* yang akan menghitung jumlah keseluruhan nomor bit bilangan biner dari angka yang diberikan.

int hitungNomorBit(int angka, int nomorBit)

Contoh:

|  |
| --- |
| // Representasi bilangan biner dari angka 13 adalah 1101  hitungNomorBit(13, 0) → mengeluarkan hasil bilangan desimal 1  hitungNomorBit(13, 1) → mengeluarkan hasil bilangan desimal 3  hitungNomorBit(13, 2) → mengeluarkan hasil null |

**Catatan Penting:**

Anda harus menulis logika sendiri untuk melakukan konversi bilangan desimal menjadi bilangan biner. Apabila menggunakan fungsi yang sudah tersedia dalam bahasa pemrograman yang digunakan maka jawaban soal akan dinyatakan tidak sah dan **mendapatkan nilai 0**.

**Bahasa Pemrograman : Javascript**

**Alamat URL Ideone.com : https://ideone.com/n3C3O4**

**Solusi :**

importPackage(java.io);

importPackage(java.lang);

var reader = new BufferedReader( new InputStreamReader(System['in']) );

const convertToBinary = (x) => {

let bin = 0;

let rem, i = 1, step = 1;

while (x !== 0) {

rem = x % 2;

x = parseInt(x / 2);

bin = bin + rem \* i;

i = i \* 10;

}

return bin.toString();

}

const hitungNomorBit = () => {

var line = reader.readLine().split(',');

const binar = convertToBinary(line[0]);

let sum = 0;

for (let i = 0; i < binar.length; i++) {

if (binar[i] == line[1]) {

sum++;

} else if (line[1] > 1){

sum = 'null';

}

}

return System.out.println(sum);

}

hitungNomorBit();

**Ujian Saringan Masuk Suitmedia**

**Bobot Soal : 35**

**Nama Kandidat : Fardan Nozami Ajitama**

**Posisi Yang Dipilih : Backend Engineer**

Buatlah sebuah *class Kamus* yang akan menyimpan semua kata sudah yang ditambahkan ke dalamnya beserta dengan sinonim dari masing-masing kata yang sudah ditambahkan.

Adapun class tersebut harus memiliki fungsi-fungsi sebagai berikut :

tambah(String kata, Array sinonim) : tidak mengembalikan hasil (*void*)

ambilSinonim(String kata) : mengembalikan hasil *array of strings*

Contoh:

|  |
| --- |
| $kamus = new Kamus();  $kamus->tambah(‘big’, [‘large’, ‘great’]);  $kamus->tambah(‘big’, [‘huge’, ‘fat’]);  $kamus->tambah(‘huge’, [‘enormous’, ‘gigantic’]);  // mengembalikan hasil [‘large’, ‘great’, ‘huge’, ‘fat’]  $kamus->ambilSinonim(‘big’);  // Perhatikan baik-baik hasil pengujian di bawah ini  // mengembalikan hasil [‘big’, ‘enormous’, ‘gigantic’]  $kamus->ambilSinonim(‘huge’);  // mengembalikan hasil [‘huge’]  $kamus->ambilSinonim(‘gigantic’);  // mengembalikan hasil null  $kamus->ambilSinonim(‘colossal’); |

**Bahasa Pemrograman :**

**Alamat URL Ideone.com :**

**Solusi :**

**Ujian Saringan Masuk Suitmedia**

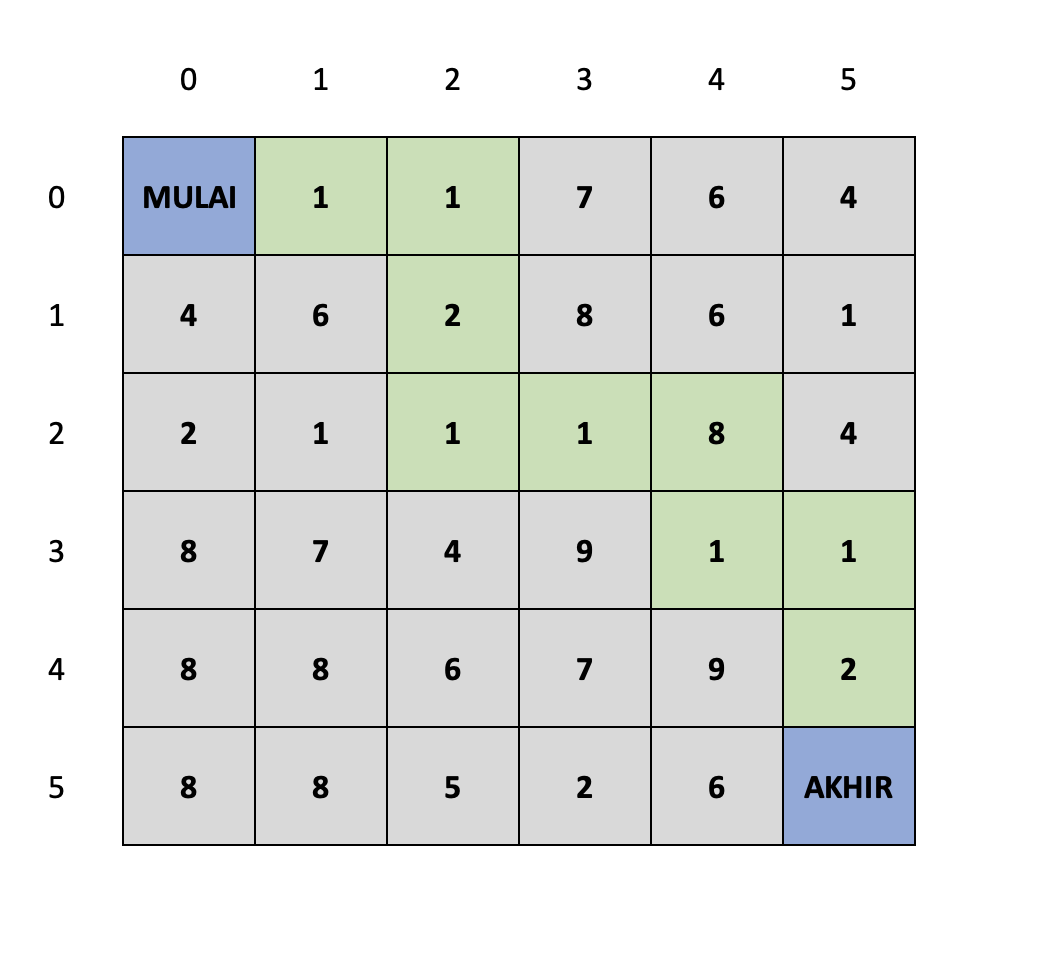
**Bobot Soal : 35**

**Nama Kandidat :**

**Posisi Yang Dipilih :**

Permainan *KotakKatik* merupakan papan permainan berupa tabel 6 x 6 yang selalu memiliki titik mulai dan titik akhir pada sudut kiri atas dan kanan bawah papan permainan. Pada kotak-kotak lainnya dalam papan permainan berisikan angka secara acak, berperan sebagai jumlah energi yang dibutuhkan untuk melewati kotak tersebut. Tujuan dari permainan ini adalah untuk menemukan jalan yang dapat menghubungkan titik awal dan titik akhir dengan jumlah total energi paling kecil.

Ilustrasi:



Jumlah energi:

1+1+2+1+1+8+1+1+2 = 18

Buatlah sebuah class KotakKatik yang memiliki dua fungsi, antara lain:

* *temukanJalan(): array*  
  Fungsi ini akan berusaha menemukan jalan yang menghubungkan titik awal dan titik akhir dengan konsumsi energi yang paling sedikit. Jalan tidak boleh diagonal.
* *totalEnergi(): int*  
  Fungsi ini akan mengembalikan nilai total energi yang dibutuhkan untuk melewati jalan yang sudah ditemukan melalui fungsi pertama. Apabila class KotakKatik belum memiliki set koordinat jalan yang menjadi solusi, maka fungsi ini akan menghasilkan error/exception.

|  |
| --- |
| class KotakKatik  {  /\*\*  \* Papan permainan  \* Koordinat [0,0] dan [5, 5] akan selalu berisikan nilai 0  \* sebagai titik awal dan titik akhir.  \*/  private array $papan = [  [0, 1, 1, 7, 6, 4],  [4, 6, 2, 8, 6, 1],  [2, 1, 1, 1, 8, 4],  [8, 7, 4, 9, 1, 1],  [8, 8, 6, 7, 9, 2],  [8, 8, 5, 2, 6, 0],  ];  private array $solusiJalan = [];  public function temukanJalan(): array  {  // Tolong implementasikan saya  }  public function totalEnergi(): int  {  // Tolong implementasikan saya  }  } |

Contoh:

|  |
| --- |
| $permainan = new KotakKatik();  // mengembalikan hasil [[0,0], [1,0], [2,0], [2,1], [2,2], [3,2], [4,2], [4,3], [5,3], [5,4], [5,5]]  $permainan->temukanJalan();  // mengembalikan hasil 18  $permainan->totalEnergi(); |