

- SECTIONS
- 1
 - 2
 - 3
 - 4
 - 5
 - 6
 - 7
 - 8

Struktur Data (E)

Dashboard / My courses / Struktur Data (E) / Pertemuan 17 / EAS Struktur Data

Question 42

Not yet answered

Marked out of 1.00

Flag question

Sebuah BST terbentuk dari deretan nilai node sbb: 10, 15, 8, 9, 13, 5, 6, 7. Hasil traversal dengan BFS pada BST tersebut adalah

Select one:

- ☐ 10, 15, 8, 9, 13, 5, 6, 7
- ☐ 10, 8, 5, 9, 15, 13, 6, 7
- ☒ 10, 8, 15, 5, 9, 13, 6, 7
- ☐ 10, 8, 5, 6, 7, 9, 15, 13

Quiz navigation

Finish attempt ...



1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36
37	38	39	40	41	42
43	44	45	46	47	48
49	50				

Previous page

Next page



Suatu problem topological sorting dengan 9 node, node-node tersebut diberi nomor/label dari 0 s/d 8, dan ada 10 persyaratan sebagai berikut:

2 -> 6 3 -> 1 7 -> 5 8 -> 4 0 -> 1

5 -> 4 1 -> 4 6 -> 7 7 -> 0 0 -> 8

Kemungkinan solusi yang didapatkan:

Select one:

☐ 3 2 6 7 5 0 8 1 4

☒ 2 6 7 5 0 8 3 1 4

☐ semua jawaban benar

☐ 2 3 6 7 0 1 5 8 4

Algoritma interpolation search merupakan variasi dari algoritma

Select one:

- ☒ Binary Search
- ☐ Jump Search
- ☐ Exponential Search
- ☐ Linear Search

[Clear my choice](#)

Question 39

Answer saved

Marked out of 2.00

Flag question

Suatu teknik pencarian data dalam array dimensi 1 yang akan menelusuri semua elemen array dari awal sampai akhir, dimana data tidak perlu diurutkan terlebih dahulu (acak) adalah metode

Select one:

- ☐ a. Linier Searching ✓
- ☒ a dan b benar
- ☐ c. Binary searching
- ☐ b. Sequential Searching ✓

[Clear my choice](#)

Diketahui algoritma sorting sebagai berikut:

```
void swap(int *xp, int *yp)
{
    int temp = *xp;
    *xp = *yp;
    *yp = temp;
}

void algo1(int arr[], int n)
{
    int i, j, min_idx;

    for (i = 0; i < n-1; i++)
    {
        min_idx = i;
        for (j = i+1; j < n; j++)
            if (arr[j] < arr[min_idx])
                min_idx = j;

        swap(&arr[min_idx], &arr[i]);
        printArray(arr, n);
    }
}
```

Jika kita memiliki input array {64, 25, 12, 22, 11}, maka isi di dalam array saat iterasi kedua (i = 1) adalah

11, 12, 25, 22, 64

Select one:

- ☐ 64, 11, 12, 22, 25
- ☐ 11, 12, 22, 25, 64
- ☐ 11, 25, 12, 22, 64



Question 37

Answer saved

Marked out of 2.00

Flag question

Jika pada potongan program berikut ini,

```
15 void insert(int value)
16 { int key = value % size;
17   if(arr[key] == -1)
18   { arr[key] = value;
19     printf("%d masuk pada arr[%d]\n", value, key);
20   }
21   else
22   { printf("Terjadi Collision : arr[%d] element %d sudah ada \n", key, arr[key]);
23     printf("Insert di batalkan | %d\n", value);
24   }
25 }
```

arr di inialisasi dengan -1, dan size=7, jika diberikan statemen-statemen insert(10); insert(4); insert(2); insert(3); dengan menggunakan metode linear probing, maka tabel hash yang akan terbentuk

Select one:

- ☒ -1,-1,2,10,4,3,-1
- ☐ -1,3,-1,10,4,2,-1
- ☐ 3,-1,2,10,4,-1,-1
- ☐ -1,2,3,10,4,-1,-1

[Clear my choice](#)

ketika insert(3) didapatkan hasilnya 3, sedangkan index 3 sudah diisi dengan 10, sehingga dilakukan linear probing hingga didapatkan index 5

Question **36**

Answer saved

Marked out of 2.00

🚩 Flag question

Diketahui fungsi hashing $H_2(\text{key}) = \text{key} \% 12 + 1$ dan akan disimpan ke dalam $\text{num}[1 \dots 12]$. Jika kita memasukkan elemen 18 dan 30, maka akan terjadi collision. Di indeks ke berapakah hasil hash angka 30 disimpan jika kita menggunakan Linear Probing (dengan $k = 1$)?

k adalah banyaknya langkah yang dilakukan dalam setiap iterasi linear probing

Select one:

☐ 10

☒ 7

☒ 8

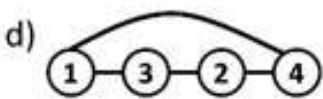
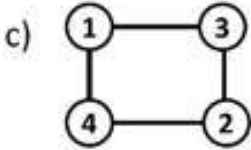
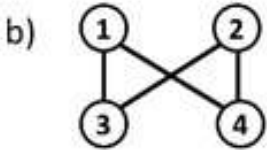
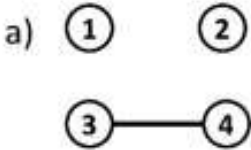
ketika insert(30) didapatkan hasilnya 7, ternyata index 7 sudah diisi dengan 18, sehingga dilakukan linear probing, dan didapatkan hasilnya 8

☐ 9

[Clear my choice](#)

Dari adjacency matriks ini, bentuk **graph** yang tidak sesuai adalah

	1	2	3	4
1	0	0	1	1
2	0	0	1	1
3	1	1	0	0
4	1	1	0	0



Select one:

☒ a)

☐ c)

☐ d)

☐ b)

[Clear my choice](#)



Potongan program dibawah ini adalah pencarian dengan interpolasi:

```

1 #include <stdio.h>
2 int interpolationSearch(int arr[], int lo, int hi, int x)
3 {
4     int pos;
5     if (lo <= hi && x >= arr[lo] && x <= arr[hi]) {
6         pos = lo
7             + (((double)(hi - lo) / (arr[hi] - arr[lo]))
8               * (x - arr[lo]));
9         if (arr[pos] == x)
10            return pos;
11        if (arr[pos] < x)
12            return interpolationSearch(arr, pos + 1, hi, x);
13        if (arr[pos] > x)
14            return interpolationSearch(arr, lo, pos - 1, x);
15    }
16    return -1;

```

inti dari program di atas adalah melakukan search, jika ditemukan maka return indexnya, jika tidak maka return -1.

Jika di fungsi main terdapat statemen:

```

int arr[] = { 10, 12, 13, 16, 18, 19, 20, 21, 22, 23, 24, 33, 35, 42, 47 };
int n = sizeof(arr) / sizeof(arr[0]);
int dicari=0;
int index = interpolationSearch(arr, 0, n - 1, dicari);

```

Berapa nilai index jika yang dicari adalah 0, 18, 50 ?

Select one:

☒ -1,4,-1

☐ 0,18,50



Question **40**

Answer saved

Marked out of 2.00

Flag question

Salah satu permasalahan yang muncul pada Linear probing adalah clustering. Masalah ini dapat diselesaikan dengan cara

Select one:

- ☒ **Menerapkan Rehashing** sudah termasuk menghitung load factor, mengubah table size, dan mungkin juga menerapkan lazy deletion
- ☐ Menghitung load factor-nya
- ☐ Menerapkan Lazy deletion
- ☐ Mengubah TableSize

[Clear my choice](#)

```
        min_idx = j;  
        swap(&arr[min_idx], &arr[i]);  
        printArray(arr, n);  
    }  
}
```

Jika kita memiliki input array {64, 25, 12, 22, 11}, maka isi di dalam array saat iterasi kedua ($i = 1$) adalah

Select one:

- ☐ 64, 11, 12, 22, 25
- ☐ 11, 12, 22, 25, 64
- ☐ 11, 25, 12, 22, 64
- ☒ 11, 12, 25, 22, 64

jawaban dari soal 33 (page 2)

[Clear my choice](#)



Question 46

Answer saved

Marked out of 2.00

Flag question

Berapa kali tercetak 'xxx' yang dilakukan oleh program dibawah ini?

main.cpp c++11.cpp

```
1 #include<stdio.h>
2 int main()
3 {   int x;
4     for(x=-1; x<=10; x++)
5     {   if(x < 5)
6         continue;
7         else
8             break;
9         printf("xxx"); }
10    return 0;
11 }
```

Select one:

☐ 11

☐ Infinite

☒ 0

☐ 10

[Clear my choice](#)



Question **43**

Answer saved

Marked out of 2.00

Flag question

Bila diketahui banyaknya ruas pada suatu **graph** = 7, maka jumlah derajat **Graph**-nya adalah

Select one:

☒ 14

☐ 49

☐ 6

☐ 21

[Clear my choice](#)

derajat suatu vertex adalah banyaknya edge yang terhubung dengan vertex tersebut

Question **44**

Answer saved

Marked out of 2.00

Flag question

Kompleksitas operasi pencarian pada hash table adalah

Select one:

- ☐ $O(\log n)$
- ☐ $O(n \log n)$
- ☒ $O(n)$
- ☐ $O(1)$

ekspektasi dari penggunaan hash table adalah setiap operasi hanya $O(1)$, tetapi realitanya masih bisa terjadi collision, sehingga worst case adalah $O(n)$

[Clear my choice](#)

Question 48

Not yet answered

Marked out of 2.00

Flag question

Berapa jumlah swapping yang diperlukan untuk mengurutkan bilangan 8,22,7,9,31,19,5,13 dari kecil ke besar menggunakan bubble sort?

Select one:

☐ 12

☐ 11

☒ 14

☐ 13

[Clear my choice](#)

jika proses swapping pada suatu sort hanya dilakukan pada angka-angka yang bersebelahan, maka untuk menghitung total swapping bisa dengan cara menghitung total inversion.

jika kita ingin melakukan sort dari kecil ke besar, maka inversion dari $arr[i]$ adalah banyaknya $arr[j]$ dengan $j > i$ tetapi $arr[j] < arr[i]$.

Berikut adalah list inversion pada array di atas

inversion 8 = 2, yaitu {7, 5}

inversion 22 = 5, yaitu {7, 9, 19, 5, 13}

inversion 7 = 1, yaitu {5}

inversion 9 = 1, yaitu {5}

inversion 31 = 3, yaitu {19, 5, 13}

inversion 19 = 2, yaitu {5, 13}

inversion 5 = 0

inversion 13 = 0

sehingga total inversion = $2 + 5 + 1 + 1 + 3 + 2 = 14$

Question **42**

Answer saved

Marked out of 2.00

Flag question

Manakah dari struktur data berikut ini yang bertipe non-linier?

Select one:

☐ stack

☐ list

☐ string

☒ graph

[Clear my choice](#)

Question **45**

Answer saved

Marked out of 2.00

🚩 Flag question

Mengelompokan deret bilangan kedalam 2 bagian, 4 bagian, 8 bagian, ... dst, merupakan salah satu iterasi didalam metode:

Select one:

☒ Merge Sort

☐ Quick Sort di QuickSort, jika suatu partisi sudah terurut, maka tidak akan lanjut dibagi-bagi lagi

☐ Selection Sort

☐ Bubble Sort

[Clear my choice](#)

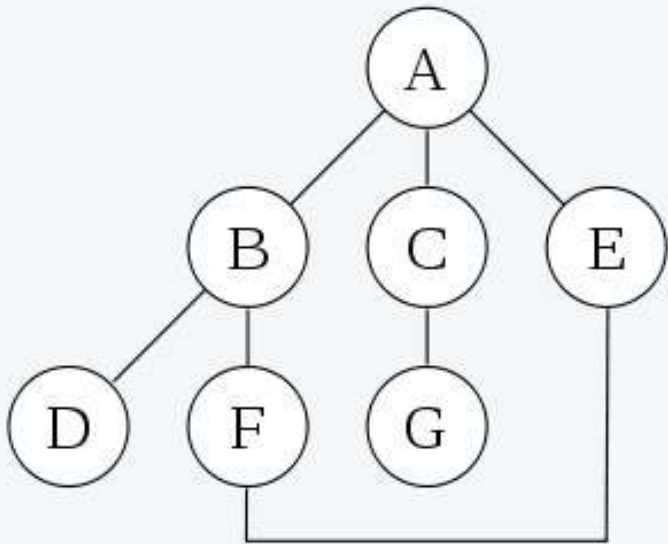
Question 47

Answer saved

Marked out of 2.00

Flag question

DFS dari **graph** berikut ini adalah



Select one:

☒ ABDFECG

☐ ABCDEFG

☐ ABCEDFG

☐ ABDFCGE

di DFS, setelah AB kemungkinannya adalah D atau F, tidak mungkin C

di DFS, setelah ABDF, satu-satunya kemungkinan adalah E, tidak mungkin C

[Clear my choice](#)



Question 50

Not yet answered

Marked out of 2.00

Flag question

Potongan program hash sederhana dibawah ini:

```
15 void insert(int value)
16 { int key = value % size;
17   if(arr[key] == -1)
18   { arr[key] = value;
19     printf("%d masuk pada arr[%d]\n", value, key);
20   }
21   else
22   { printf("Terjadi Collision : arr[%d] element %d sudah ada \n", key, arr[key]);
23     printf("Insert di batalkan | %d\n", value);
24   }
25 }
```

Jika arr di inisialisasi dengan -1, dan size=7, maka statemen insert(10); insert(4); insert(2); insert(3); akan terbentuk hash tabel yang tepat adalah?

Select one:

☒ -1,2,10,4,-1,-1,-1

☐ -1,-1,2,10,4,-1,-1

☐ -1,3,2,10,4,-1,-1

☐ -1,2,3,10,4,-1,-1

[Clear my choice](#)

Ketika insert(3) didapatkan hasilnya adalah 3, sedangkan index 3 sudah diisi dengan 10. Karena tidak dilakukan linear probing ketika terjadi collision, maka angka 3 tidak jadi diinsert ke dalam hash table.

Question **49**

Answer saved

Marked out of 2.00

🚩 Flag question

Suatu perusahaan memiliki cabang di seluruh Indonesia. Salah seorang supervisor diberi tugas mengunjungi semua cabang perusahaan tersebut dengan dibekali alamat dan jarak dari satu cabang ke cabang lainnya serta diberi uang perjalanan. Jika ada uang perjalanan yang tersisa, maka bisa diambil sebagai bonus. Cabang-cabang perusahaan direpresentasikan sebagai *vertex/node* dan jarak antar cabang direpresentasikan sebagai *edge* dalam suatu *graph*. Untuk mendapatkan bonus terbesar, implementasi apa yang dapat dilakukan oleh supervisor tersebut?

Select one:

- ☐ Shortest path
- ☒ Minimum spanning tree
- ☐ Breadth-first traversal
- ☐ Depth-first traversal

[Clear my choice](#)

Question **41**

Answer saved

Marked out of 2.00

Flag question

Dari keempat algoritma sorting berikut, manakah yang menggunakan pendekatan divide-and-conquer? 1. Bubble sort 2. Quick sort 3. Merge sort 4. Selection sort

Select one:

☒ 2 dan 3

☐ Semua dapat menggunakan divide-and-conquer

☐ 3 dan 4

☐ 1 dan 2

[Clear my choice](#)

Divide and Conquer adalah teknik memecah problem menjadi beberapa sub-problem, kemudian setiap sub-problem akan dikerjakan masing-masing untuk mendapatkan solusi dari masalah secara keseluruhan.

MergeSort dan QuickSort melakukan pembagian array menjadi beberapa bagian (sub-array), kemudian dilakukan sort pada masing-masing bagian, sehingga pada akhirnya didapatkan hasil array yang sudah disort.

Question **1**

Answer saved

Marked out of 2.00

Flag question

Apakah output program dibawah ini, jika di angka 25 dimasukkan sebagai input?

```
1 #include<stdio.h>
2 int main()
3 {   int i;  printf("%d\n", scanf("%d", &i));  return 0;}
4 |
```

Select one:

☐ 5

☒ 1

☐ 25

☐ 2

[Clear my choice](#)

terdapat 3 jenis return value dari function scanf :

- 1) jika scanf berhasil menerima input, maka return banyaknya variabel yang berhasil diinput
- 2) jika scanf gagal menerima input (bisa jadi karena error), maka return 0
- 3) jika input stream habis, maka return EOF (End Of File)

Potongan program di bawah ini:

```
1 #include<stdio.h>
2 #define V 2
3 void init(int arr[][V])
4 {   int i,j;
5     for(i = 0; i < V; i++)
6         for(j = 0; j < V; j++)
7             arr[i][j] = 0;}
8 void addEdge(int arr[][V],int src, int dest)
9 {   arr[src][dest] = 1;}
10
11 void printAdjMatrix(int arr[][V])
12 {   int i, j;
13     for(i = 0; i < V; i++)
14     {   for(j = 0; j < V; j++)
15         { printf("%d ", arr[i][j]);}
16         printf("\n");}
17 }
```

Apakah yang terjadi jika main program seperti di bawah ini?

```
int main() {
    int adjMatrix[V][V];init(adjMatrix);
    addEdge(adjMatrix,0,1);
    addEdge(adjMatrix,0,2);
    addEdge(adjMatrix,0,3);
    addEdge(adjMatrix,1,3);
    addEdge(adjMatrix,1,4);
    addEdge(adjMatrix,2,3);
```



```

int main() {
    int adjMatrix[V][V]; init(adjMatrix);
    addEdge(adjMatrix, 0, 1);
    addEdge(adjMatrix, 0, 2);
    addEdge(adjMatrix, 0, 3);
    addEdge(adjMatrix, 1, 3);
    addEdge(adjMatrix, 1, 4);
    addEdge(adjMatrix, 2, 3);
    addEdge(adjMatrix, 3, 4);
    printAdjMatrix(adjMatrix);
    return 0;
}

```

Select one:

Program error (berhenti)

☒ 0 1 1 1 0
0 0 0 1 1
0 0 0 1 0
0 0 0 0 1
0 0 0 0 0

HATI-HATI !!!

Ukuran matrix hanyalah $V \times V$, sedangkan nilai V adalah 2, sehingga ukuran matrix hanyalah 2×2 .
Sehingga ketika addEdge vertex yang nomornya ≥ 2 , akan terjadi error, contohnya addEdge(2, 3)

☐ 0 1 1 1 1
0 0 0 1 1
0 0 0 1 0
0 0 0 0 1
0 0 0 0 0

☐ 1 1 1 1 1
0 0 0 1 1
0 0 0 1 0
0 0 0 0 1
0 0 0 0 0



Potongan program Sorting berikut ini:

```
1 void QuickSort(int array[],int start,int end)
2 {int index=start,i,pivot=array[end],temp;
3  if(start<end) {
4      for(i=start;i<end;i++)
5          {if(array[i]<=pivot)
6              {temp=array[i];
7                array[i]=array[index];
8                array[index]=temp;
9                index++;}}
10     }
11     temp=array[index];
12     array[index]=array[end];
13     array[end]=temp;
14     QuickSort(array,start,index-1);
15     QuickSort(array,index+1,end);
16 }
```

Bagaimana cara memanggil fungsi diatas jika di ketahui num adalah jumlah datanya dan array adalah datanya?

Select one:

- ☒ QuickSort(array,0,num-1)
- ☐ QuickSort(&array,0,num)
- ☐ QuickSort(&array,0,num-1)
- ☐ QuickSort(array,0,num)

[Clear my choice](#)



Question **23**

Answer saved

Marked out of 2.00

🚩 Flag question

Misalkan kita mengurutkan array delapan bilangan bulat menggunakan quicksort, dan kita baru saja menyelesaikan partisi pertama dengan array yang terlihat seperti ini: 2 5 1 7 9 12 11 10.

Pernyataan mana yang benar?

Select one:

- ☐ Baik angka 7 maupun angka 9 bukanlah pivot
- ☐ Pivotnya bisa jadi 7, tapi bukan 9
- ☐ Pivotnya bukan 7, tapi bisa jadi 9
- ☒ Pivotnya bisa berupa 7 atau 9.

[Clear my choice](#)

semua angka di sebelah kiri 7 nilainya ≤ 7 , dan semua angka di sebelah kanan 7 nilainya > 7 , sehingga 7 bisa jadi pivot.

semua angka di sebelah kiri 9 nilainya ≤ 9 , dan semua angka di sebelah kanan 9 nilainya > 9 . sehingga 9 bisa jadi pivot.

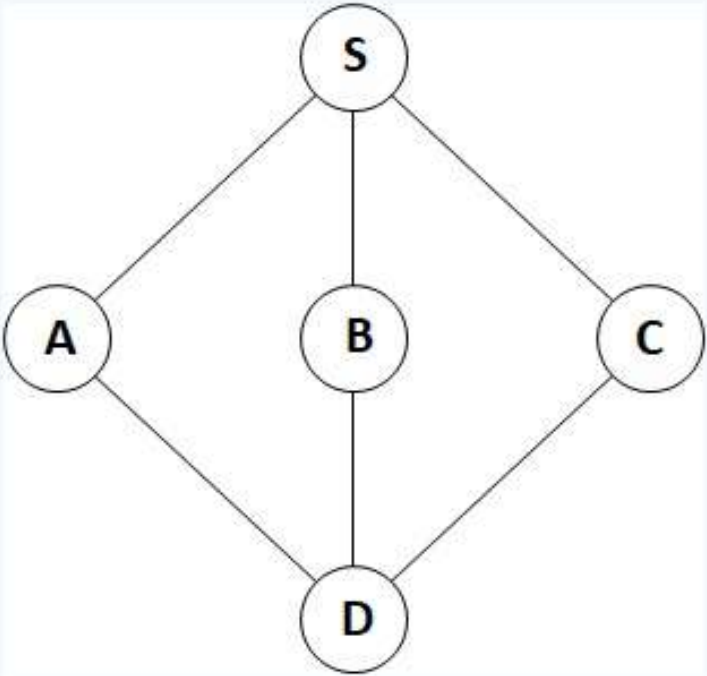
Question **21**

Answer saved

Marked out of 2.00

Flag question

Hasil Breadth First Search dari **graph** berikut ini yang dimulai dari node S adalah



Select one:

- ☐ SADBC
- ☐ ASBDC
- ☐ ASBCD
- ☒ SABCD

jika BFS, setelah S seharusnya A atau B atau C, sedangkan D dikunjungi setelah ABC sudah dikunjungi semua

[Clear my choice](#)



Question 32

Answer saved

Marked out of 2.00

Flag question

Tabel hash dengan panjang 10 menggunakan pengalamatan terbuka (open addressing) dengan fungsi hash $h(k)=k \bmod 10$, dan probing linier. Setelah memasukkan 6 nilai ke dalam tabel hash kosong, tabelnya seperti gambar di bawah ini:

0	
1	
2	42
3	23
4	34
5	52
6	46
7	33
8	
9	

Manakah dari pilihan berikut yang memberikan urutan memasukkan nilai sehingga mendapatkan hasil seperti table di atas?

Select one:

- ☐ 34, 42, 23, 52, 33, 46 hasil = { -, -, 42, 23, 35, 52, 33, 46, -, - }
- ☒ 46, 42, 34, 52, 23, 33 hasil = { -, -, 42, 52, 34, 23, 46, 33, -, - }
- ☐ 46, 34, 42, 23, 52, 33 hasil = { -, -, 42, 23, 34, 52, 46, 33, -, - }
- ☐ 42, 46, 33, 23, 34, 52 hasil = { -, -, 42, 33, 23, 34, 46, 52, -, - }

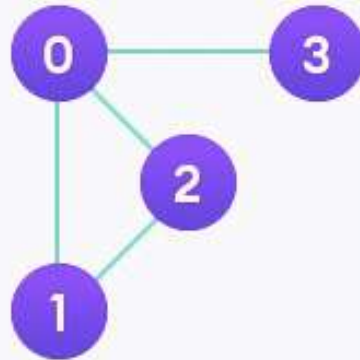
Question 8

Answer saved

Marked out of 2.00

Flag question

Terdapat **graph** seperti dibawah ini:



Bagaimanakah representasi dalam adjacent list?

Select one:

- ☐ 0->1->2->3
1->0->2
2->0->1
3->0->1
- ☒ 0->1->2->3
1->0->2
2->0->1
3->0
- ☐ 0->1->0->3
1->2->0->3
2->0->1
3->0
- ☐ 0->1->2->0->3
1->2->0->3
2->0->1
3->0



Question 2

Answer saved

Marked out of 2.00

Flag question

Teknik pengurutan data yang paling tepat digunakan untuk tabel data yang sangat besar adalah

Select one:

☐ Bubble Sort

☐ Quick Sort

☐ Selection Sort

☒ Merge Sort

BubbleSort dan SelectionSort kompleksitasnya $O(n^2)$
QuickSort dan MergeSort kompleksitasnya $O(n \cdot \log n)$

sehingga BubbleSort dan SelectionSort tidak efektif untuk tabel ukuran besar

MergeSort lebih stabil / konsisten dibandingkan QuickSort

[Clear my choice](#)

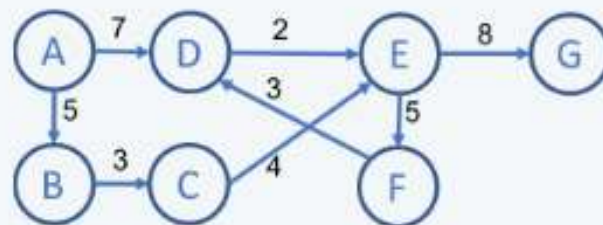
Question 24

Answer saved

Marked out of 2.00

Flag question

Diketahui jarak (dalam km) antar kota A, B, C, D, E, F, dan G sebagai berikut:



Pilihan metode tercepat untuk mendapatkan jarak terpendek dari kota A ke kota G adalah

Select one:

☐ Depth-First Search (20 km)

DFS tidak bisa digunakan untuk mencari shortest path

☒ Breadth-First Search (17 km)

☐ Binary Search (17 km)

Binary Search tidak digunakan untuk Graph

☐ Breadth-First Search (20 km)

[Clear my choice](#)

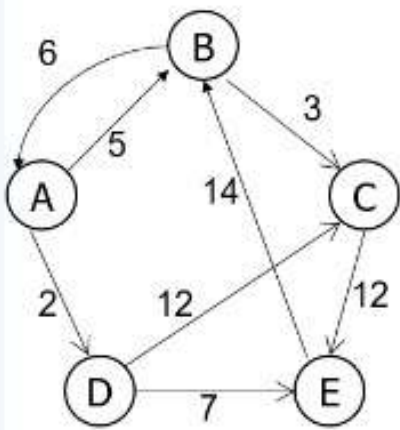
Question 26

Answer saved

Marked out of 2.00

Flag question

Digambarkan **graph** berarah dan berbobot seperti di bawah ini:



	A	B	C	D	E
A 0	0	5	0	2	0
B 1	6	0	3	0	0
C 2	0	0	0	0	9
D 3	0	0	12	0	7
E 4	0	14	0	0	0

Pilihlah yang **SALAH** dari representasi matrix di bawah ini:

Select one:

☒ C->E

☐ E->B

☐ Semua jawaban salah

☐ A->D

[Clear my choice](#)

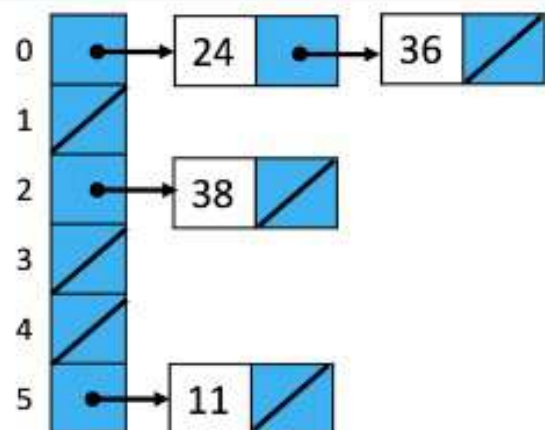
Question 14

Answer saved

Marked out of 2.00

Flag question

Diketahui chaining hashing dengan struktur sebagai berikut:



Jika struktur data tersebut diubah menggunakan metode Linear probing dengan rumus $(x + 3) \% \text{TableSize}$, maka representasi yang benar adalah

Select one:

☒ {24, _, 38, 36, _, 11}

☐ {36, _, 38, 24, _, 11}

☐ {24, 36, 38, _, _, 11}

☐ {36, 24, 38, _, _, 11}

[Clear my choice](#)

berdasarkan gambar chaining hashing, 36 diinsert setelah 24, sehingga ketika dijalankan Linear Probing, 36 akan dipindah ke index 3.

Question 6

Answer saved

Marked out of 2.00

Flag question

Diketahui algoritma searching sebagai berikut:

```
int search(int array[], int n, int x)
{
    for (int i = 0; i < n; i++)
    {
        if (array[i] == x)
            return i;
    }

    return -1;
}
```

Kita memiliki array {2, 4, 0, 1, 9} dan mencari data 0, 4, dan 5 secara berurutan. Maka kembalian indeks dari hasil pencarian data tersebut secara berurutan adalah

Select one:

☐ 0, 4, 5☒ 2, 1, -1☐ 0, 4, -1☐ 0, 1, 9[Clear my choice](#)

Question **29**

Answer saved

Marked out of 2.00

Flag question

Berapa banyak perbandingan yang diperlukan untuk mengurutkan sebuah array yang mengandung 10 elemen menggunakan Selection Sort jika array tersebut komposisi awalnya sudah terurut?

Select one:

☐ 10

☐ 9

☐ 23

☒ 45

[Clear my choice](#)

Loop i dijalankan dari $i = 0$ hingga $i = 8$

ketika $i = 0$, dijalankan loop j dari $j = 1$ hingga $j = 9$

ketika $i = 1$, dijalankan loop j dari $j = 2$ hingga $j = 9$

...

ketika $i = 8$, dijalankan loop j dari $j = 9$ hingga $j = 9$

Karena dilakukan proses perbandingan pada setiap loop j, maka total perbandingan yang dilakukan adalah

$$9 + 8 + \dots + 1 = 45$$

topological sorting adalah suatu problem pada directed graph, dimana untuk mengunjungi suatu vertex, kita harus mengunjungi seluruh vertex lain yang mengarah ke vertex tersebut dulu.

Contoh :

untuk mengikuti matkul OOP, harus lulus matkul StrukDat

untuk mengikuti matkul APSI, harus lulus matkul OOP

untuk mengikuti matkul APSI, harus lulus matkul SBD

sehingga topological sortingnya adalah :

{ StrukDat, SBD, OOP, APSI }

Question 34

Answer saved

Marked out of 2.00

Flag question

Representasi **graph** dari suatu problem topological sorting bisa mempunyai solusi jika graphnya adalah

Select one:

☐ cyclicgraphdan acyclicgraph

☒ acyclicgraph

☐ bukan merupakan cyclicgraphdan acyclicgraph

☐ cyclicgraph

[Clear my choice](#)

Topological Sorting tidak akan bekerja pada cyclic graph, contoh :

untuk ke A, harus ke B dulu

untuk ke B, harus ke C dulu

untuk ke C, harus ke A dulu

sehingga kita tidak bisa menentukan vertex mana dulu yang harus dikunjungi

Answer saved

Question **35**

Answer saved

Marked out of 2.00

🚩 Flag question

Pada algoritma quicksort, satu nilai dipilih untuk menjadi pembatas antara nilai yang lebih rendah dan nilai yang lebih tinggi sehingga kelompok data dapat terbagi dua. Nilai tersebut disebut

Select one:

☒ Pivot

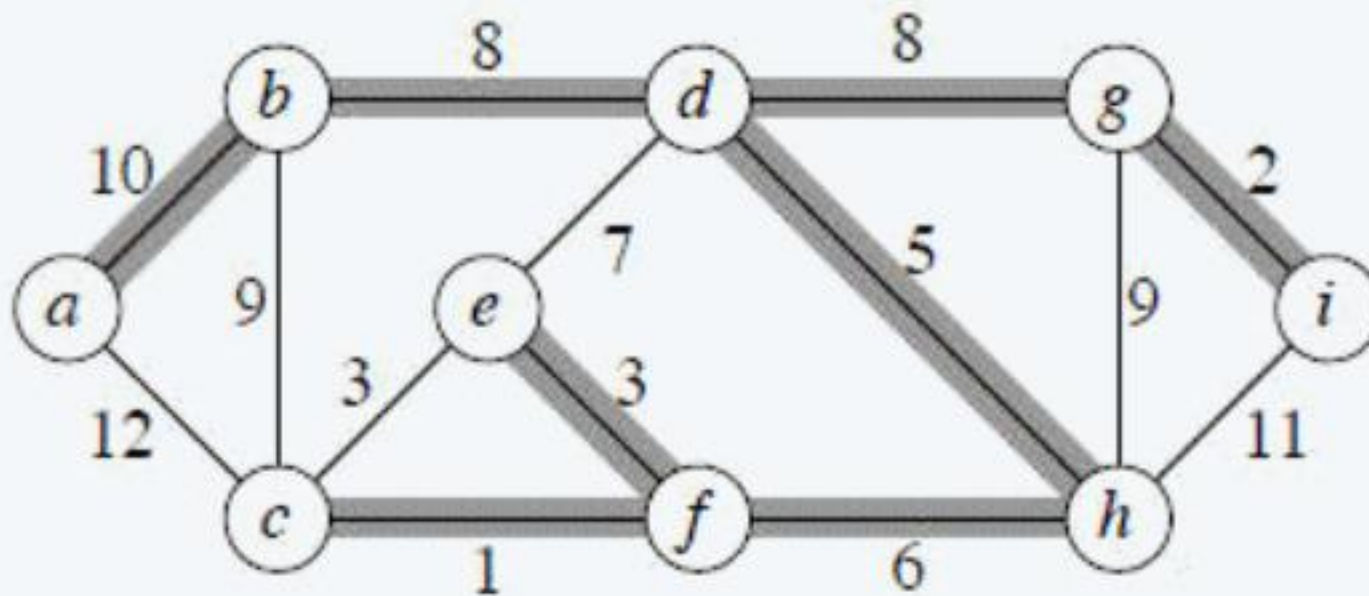
☐ Stack

☐ Swap point

☐ Stable point

[Clear my choice](#)

Garis tebal pada graph di bawah ini adalah minimum-cost spanning tree (MST), yang diperoleh dengan algoritma



Select one:

- ☐ Pre-order traversal
- ☐ Kruskal
- ☐ AVL
- ☒ Prim



Recent ▾

myITS
classroom

Not yet answered

Marked out of 1.00

Flag question

Sebuah BST terbentuk dari deretan nilai node sbb: 10, 15, 8, 9, 13, 5, 6, 7. Apa hasil pre-order traversal setelah proses balancing?

Select one:

- ☐ 5, 7, 6, 9, 8, 13, 15, 10
- ☒ 10, 8, 6, 5, 7, 9, 15, 13
- ☐ 5, 6, 7, 8, 9, 10, 13, 15
- ☐ 10, 8, 15, 6, 9, 13, 5, 7

[Clear my choice](#)

Previous page

Next page

Previous Activity

Suatu problem topological sorting dengan 9 node, diberi node-node tersebut nomor/label dari 0 s/d 8, dan ada 10 persyaratan sebagai berikut:

2 -> 6 3 -> 1 7 -> 5 8 -> 4 0 -> 1

5 -> 4 1 -> 4 6 -> 7 7 -> 0 0 -> 8

Jika ditambahkan satu syarat lagi 4 -> 2, maka

Select one:

- ☐ tidak ada solusi untuk problem topological sort tersebut
- ☐ semua jawaban salah
- ☒ tetap ada solusi untuk problem topological sort tersebut
- ☐ tidak ada pengaruhnya penambahan persyaratan tersebut

Question **29**

Answer saved

Marked out of 1.00

🚩 Flag question

Jawaban berikut ini merupakan operasi-operasi yang ada pada hash table, kecuali

Select one:

☐ Remove

☐ Find

☒ Push

☐ Insert

[Clear my choice](#)

- SECTIONS
- 1
 - 2
 - 3
 - 4
 - 5
 - 6
 - 7
 - 8

Struktur Data (E)

Dashboard / My courses / Struktur Data (E) / Pertemuan 17 / EAS Struktur Data

Question **33**
Not yet answered
Marked out of 1.00
Flag question

Diketahui fungsi hashing $H_2(\text{key}) = \text{key} \% 12 + 1$ dan akan disimpan ke dalam $\text{num}[1...12]$. Jika kita memasukkan elemen 18 dan 30, maka akan terjadi collision. Di indeks ke berapakah hasil hash angka 30 disimpan jika kita menggunakan Quadratic Probing (dengan persamaan $i + i^2$, dan $i = 1$)?

- Select one:
- ☐ 7
 - ☐ 10
 - ☒ 9
 - ☐ 8

Quiz navigation

Finish attempt ...



1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36
37	38	39	40	41	42
43	44	45	46	47	48
49	50				

Previous page

Next page

Diketahui fungsi hashing $H_2(\text{key}) = \text{key} \% 12 + 1$ dan akan disimpan ke dalam $\text{num}[1...12]$. Jika kita memasukkan elemen 18 dan 30, maka akan terjadi collision. Di indeks ke berapakah hasil hash angka 30 disimpan jika kita menggunakan Linear Probing (dengan $k = 1$)?

Select one:

- ☐ 7
- ☐ 9
- ☐ 10
- ☒ 8

- SECTIONS
- 1
 - 2
 - 3
 - 4
 - 5
 - 6
 - 7
 - 8

Struktur Data (E)

Dashboard / My courses / Struktur Data (E) / Pertemuan 17 / EAS Struktur Data

Question 35
Not yet answered
Marked out of 1.00
Flag question

Nilai- nilai integer dimasukkan ke dalam tabel hash $H[1..11]$ menggunakan fungsi hash utama $h1(k) = 1 + k \bmod 11$.
Tunjukkan status array setelah memasukkan kunci 10, 22, 31, 4, 15, 28, 17, 88, dan 58 menggunakan **double hashing** dengan $h2(k) = 1 + k \bmod 9$.

- Select one:
- ☐ 22, 88, - , 58, 4, 15, 28, 17, - , 31, 10
 - ☐ 22, 17, 88, 58, 4, - , 15, - , 28, 31, 10
 - ☐ 22, 15, - , 58, 4, 17, 28, - , 88, 31, 10
 - ☒ Semua jawaban salah

Quiz navigation

Finish attempt ...



1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36
37	38	39	40	41	42
43	44	45	46	47	48
49	50				

Previous page

Next page

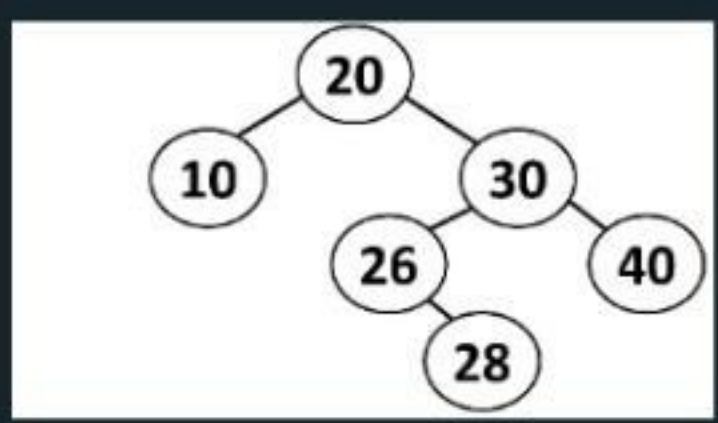
- SECTIONS
- 1
 - 2
 - 3
 - 4
 - 5
 - 6
 - 7
 - 8

Struktur Data (E)

Dashboard / My courses / Struktur Data (E) / Pertemuan 17 / EAS Struktur Data

Question 36
Not yet answered
Marked out of 1.00
Flag question

Hasil traverse nilai BST pada gambar berikut ini dengan Breath-First Search adalah



- Select one:
- ☐ 10, 20, 26, 28, 30, 40
 - ☒ 10, 20, 30, 26, 40, 28
 - ☐ 10, 40, 30, 28, 20, 26
 - ☐ 20, 10, 30, 26, 40, 28

[Clear my choice](#)

Quiz navigation

Finish attempt ...


Progress indicators for HOURS (0), MINUTES (27), and SECONDS (49).

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36
37	38	39	40	41	42
43	44	45	46	47	48
49	50				

Question **13**

Answer saved

Marked out of 1.00

 Remove flag

Manakah dari algoritma pengurutan berikut yang memberikan kinerja terbaik ketika diterapkan pada array dengan data yang sudahurut atau hampir urut (maksimum 1 atau dua elemen yang masih salah tempat)?

Select one:

- ☒ Quick Sort
- ☐ Merge Sort
- ☐ Heap Sort
- ☐ Insertion Sort

[Clear my choice](#)

- SECTIONS
- 1
 - 2
 - 3
 - 4
 - 5
 - 6
 - 7
 - 8

Struktur Data (E)

Dashboard / My courses / Struktur Data (E) / Pertemuan 17 / EAS Struktur Data

Question 43
Not yet answered
Marked out of 1.00
Flag question

Kemungkinan terbaik (best case) untuk pencarian data dengan sequential search adalah jika

- Select one:
- ☐ Data yang dicari tidak ada pada indeks array
 - ☒ Data yang dicari berada di indeks terdepan
 - ☐ Data yang dicari berada di indeks belakang
 - ☐ Data yang dicari berada di tengah indeks

Previous page
Previous Activity

Jump to...

Next page

Quiz navigation

Finish attempt ...



1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36
37	38	39	40	41	42
43	44	45	46	47	48
49	50				

Question 2

Not yet answered

Marked out of 1.00

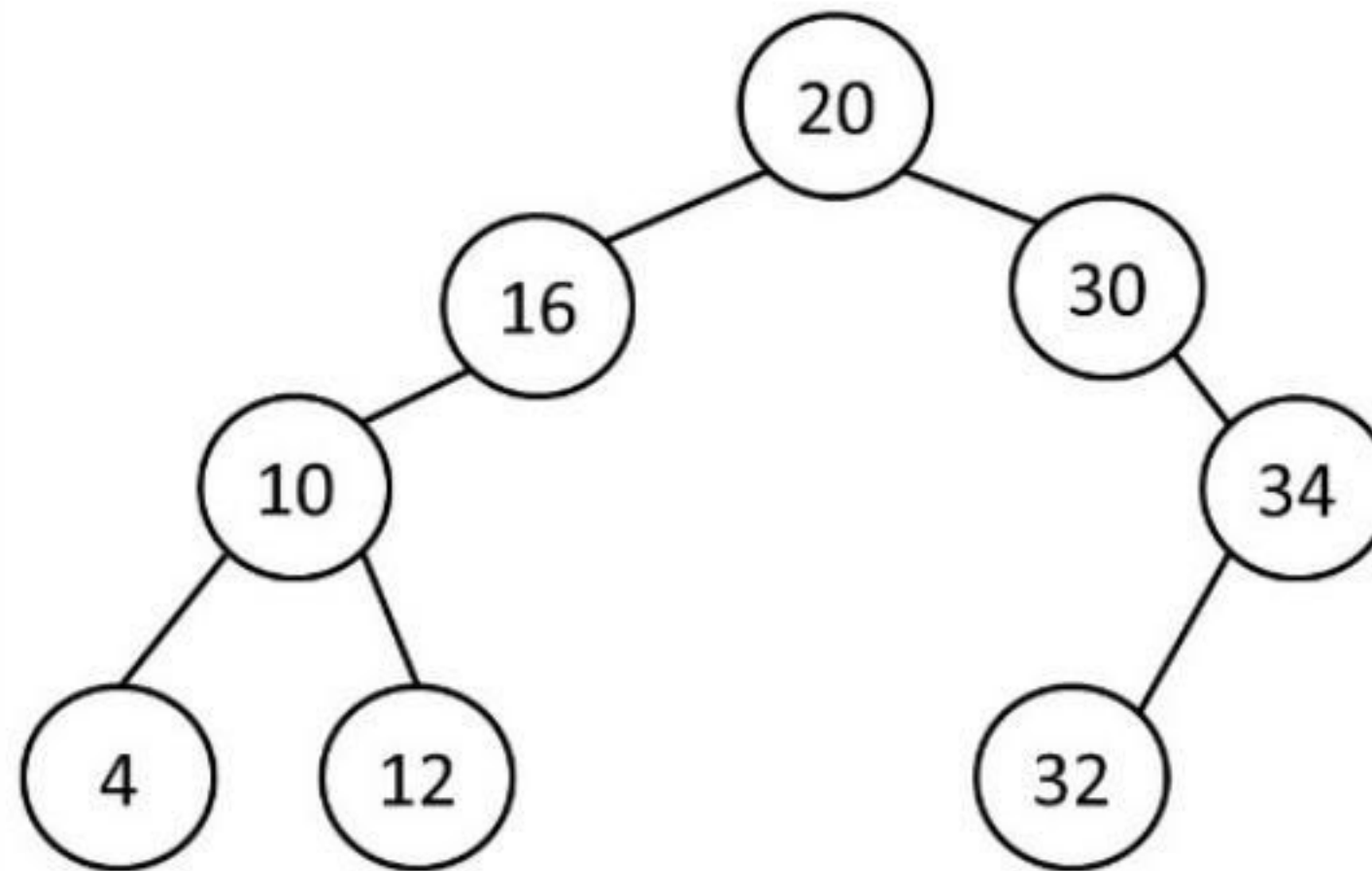
Flag question

Nilai-nilai integer di-input-kan ke dalam tabel hash **H[1..11]** menggunakan fungsi hash utama **$h_1(k) = k \% 11 + 1$** . Status array setelah memasukkan kunci 10, 22, 31, 4, 15, 28, 17, 88, dan 58 menggunakan **linear probing**:

Select one:

- ☐ Semua jawaban salah
- ☐ 22, 17, 88, - , 4, 58 , 15, - , 28, 31, 10
- ☐ 22, 17, 88, 58, 4, - , 15, - , 28, 31, 10
- ☒ 22, 88, - , 58, 4, 15, 28, 17, - , 31, 10





Select one:

- ☐ Tidak seimbang, dapat diseimbangkan dengan proses Single-Right-Rotation
- ☐ Tidak seimbang, dapat diseimbangkan dengan proses Single-Left-Rotation
- ☒ Tidak seimbang, dapat diseimbangkan dengan proses double rotation
- ☐ Tidak memerlukan rotasi (sudah seimbang)

Marked out of 1.00

Flag question

SECTIONS

1

2

3

4

5

6

7

8



- Select one:
- ☒ Separate chaining
 - ☐ Double hashing
 - ☐ Quadratic probing
 - ☐ Linear Probing

Quiz navigation

Finish attempt ...

HOURS 0 MINUTES 12 SECONDS 40

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36
37	38	39	40	41	42
43	44	45	46	47	48
49	50				

Question 20

Not yet answered

Marked out of 1.00

Flag question

Suatu *balance (AVL) binary search tree* yang pada awalnya merupakan *tree* yang kosong, kemudian di-input-kan data (keys) berikut ini:

mac tee ode era ria lea vim

secara berurutan. Setiap input data ke dalam *tree* tersebut harus merupakan *AVL tree* sebelum data berikutnya di-input-kan. Apakah hasil *in-order traversal* dari *binary search tree* tersebut?

Select one:

- ☒ era lea mac ode ria tee vim
- ☐ era mac lea ria vim tee ode
- ☐ ode lea era mac tee ria vim
- ☐ ode lea tee era mac ria vim

Question **31**

Not yet answered

Marked out of 1.00

🚩 Flag question

Diketahui array dengan urutan sebagai berikut: 57, 48, 23, 12, 39. Jika array tersebut diurutkan menggunakan Insertion Sort, maka pada akhir iterasi kedua susunannya adalah

Select one:

- ☐ 12, 48, 23, 57, 39
- ☐ 23, 48, 39, 12, 57
- ☒ 23, 48, 57, 12, 39
- ☐ 12, 23, 48, 57, 39

- SECTIONS
- 1
 - 2
 - 3
 - 4
 - 5
 - 6
 - 7
 - 8

Struktur Data (E)

Dashboard / My courses / Struktur Data (E) / Pertemuan 17 / EAS Struktur Data

Question 9
Not yet answered
Marked out of 1.00
Flag question

Nilai-nilai integer dimasukkan ke dalam tabel hash $H[1..11]$ menggunakan fungsi hash utama $h_1(k) = k \bmod 11 + 1$.
Tunjukkan status array setelah memasukkan kunci 10, 22, 31, 4, 15, 28, 17, 88, dan 58 menggunakan **quadratic probing** dengan *probe function* $i + i^2$.

- Select one:
- ☐ 22, 88, - , 58, 4, 15, 28, 17, - , 31, 10
 - ☐ 22, 17, 88, 58, 4, - , 15, - , 28, 31, 10
 - ☒ Semua jawaban salah
 - ☐ 22, 88, 17, 58, 4, - , 15, - , 28, 31, 10

Quiz navigation

Finish attempt ...



1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36
37	38	39	40	41	42
43	44	45	46	47	48
49	50				

Previous page

Next page