

Rajalakshmi Engineering College

Name: vasugi e.v.n
Email: 240801370@rajalakshmi.edu.in
Roll no: 240801370
Phone: 7708989508
Branch: REC
Department: I ECE AF
Batch: 2028
Degree: B.E - ECE

Scan to verify results



NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 1_MCQ

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : MCQ

1. Linked lists are not suitable for the implementation of?

Answer

Binary search

Status : Correct

Marks : 1/1

2. Consider an implementation of an unsorted singly linked list. Suppose it has its representation with a head pointer only. Given the representation, which of the following operations can be implemented in $O(1)$ time?

- i) Insertion at the front of the linked list
- ii) Insertion at the end of the linked list
- iii) Deletion of the front node of the linked list

iv) Deletion of the last node of the linked list

Answer

I and III

Status : Correct

Marks : 1/1

3. Given a pointer to a node X in a singly linked list. If only one point is given and a pointer to the head node is not given, can we delete node X from the given linked list?

Answer

Possible if X is not last node.

Status : Correct

Marks : 1/1

4. The following function takes a singly linked list of integers as a parameter and rearranges the elements of the lists.

The function is called with the list containing the integers 1, 2, 3, 4, 5, 6, 7 in the given order. What will be the contents of the list after the function completes execution?

```
struct node {  
    int value;  
    struct node* next;  
};
```

```
void rearrange (struct node* list) {  
    struct node *p,q;  
    int temp;  
    if (! List || ! list->next) return;  
    p=list; q=list->next;  
    while(q) {  
        temp=p->value; p->value=q->value;  
        q->value=temp;p=q->next;  
        q=p?p->next:0;  
    }  
}
```

}

Answer

2, 1, 4, 3, 6, 5, 7

Status : Correct

Marks : 1/1

5. Consider the singly linked list: 13 -> 4 -> 16 -> 9 -> 22 -> 45 -> 5 -> 16 -> 6, and an integer K = 10, you need to delete all nodes from the list that are less than the given integer K.

What will be the final linked list after the deletion?

Answer

13 -> 16 -> 22 -> 45 -> 16

Status : Correct

Marks : 1/1

6. Given the linked list: 5 -> 10 -> 15 -> 20 -> 25 -> NULL. What will be the output of traversing the list and printing each node's data?

Answer

5 10 15 20 25

Status : Correct

Marks : 1/1

7. In a singly linked list, what is the role of the "tail" node?

Answer

It stores the last element of the list

Status : Correct

Marks : 1/1

8. Consider the singly linked list: 15 -> 16 -> 6 -> 7 -> 17. You need to delete all nodes from the list which are prime.

What will be the final linked list after the deletion?

Answer

15 -> 16 -> 6

Status : Correct

Marks : 1/1

9. The following function reverse() is supposed to reverse a singly linked list. There is one line missing at the end of the function.

What should be added in place of "/*ADD A STATEMENT HERE*/", so that the function correctly reverses a linked list?

```
struct node {  
    int data;  
    struct node* next;  
};  
static void reverse(struct node** head_ref) {  
    struct node* prev = NULL;  
    struct node* current = *head_ref;  
    struct node* next;  
    while (current != NULL) {  
        next = current->next;  
        current->next = prev;  
        prev = current;  
        current = next;  
    }  
    /*ADD A STATEMENT HERE*/  
}
```

Answer

*head_ref = prev;

Status : Correct

Marks : 1/1

10. Which of the following statements is used to create a new node in a singly linked list?

```
struct node {  
    int data;
```

```
    struct node * next;  
}  
typedef struct node NODE;  
NODE *ptr;
```

Answer

```
ptr = (NODE*)malloc(sizeof(NODE));
```

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: vasugi e.v.n
Email: 240801370@rajalakshmi.edu.in
Roll no: 240801370
Phone: 7708989508
Branch: REC
Department: I ECE AF
Batch: 2028
Degree: B.E - ECE

Scan to verify results



NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 3_MCQ_Updated

Attempt : 1
Total Mark : 20
Marks Obtained : 0

Section 1 : MCQ

1. Which of the following operations allows you to examine the top element of a stack without removing it?

Answer

Status : Skipped

Marks : 0/1

2. A user performs the following operations on stack of size 5 then which of the following is correct statement for Stack?

```
push(1);  
pop();  
push(2);  
push(3);  
pop();
```

```
push(2);  
pop();  
pop();  
push(4);  
pop();  
pop();  
push(5);
```

Answer

-

Status : -

Marks : 0/1

3. When you push an element onto a linked list-based stack, where does the new element get added?

Answer

-

Status : -

Marks : 0/1

4. What will be the output of the following code?

```
#include <stdio.h>  
#define MAX_SIZE 5  
void push(int* stack, int* top, int item) {  
    if (*top == MAX_SIZE - 1) {  
        printf("Stack Overflow\n");  
        return;  
    }  
    stack[++(*top)] = item;  
}  
int pop(int* stack, int* top) {  
    if (*top == -1) {  
        printf("Stack Underflow\n");  
        return -1;  
    }  
    return stack[(*top)--];  
}
```

```
}
```

```
int main() {  
    int stack[MAX_SIZE];  
    int top = -1;  
    push(stack, &top, 10);  
    push(stack, &top, 20);  
    push(stack, &top, 30);  
    printf("%d\n", pop(stack, &top));  
    printf("%d\n", pop(stack, &top));  
    printf("%d\n", pop(stack, &top));  
    printf("%d\n", pop(stack, &top));  
    return 0;  
}
```

Answer

-

Status : -

Marks : 0/1

5. What will be the output of the following code?

```
#include <stdio.h>  
#define MAX_SIZE 5  
int stack[MAX_SIZE];  
int top = -1;  
void display() {  
    if (top == -1) {  
        printf("Stack is empty\n");  
    } else {  
        printf("Stack elements: ");  
        for (int i = top; i >= 0; i--) {  
            printf("%d ", stack[i]);  
        }  
        printf("\n");  
    }  
}  
void push(int value) {  
    if (top == MAX_SIZE - 1) {
```



```

        printf("Stack Overflow\n");
    } else {
        stack[++top] = value;
    }
}
int main() {
    display();
    push(10);
    push(20);
    push(30);
    display();
    push(40);
    push(50);
    push(60);
    display();
    return 0;
}

```

Answer

-

Status : -

Marks : 0/1

6. Which of the following Applications may use a Stack?

Answer

-

Status : -

Marks : 0/1

7. Elements are Added on _____ of the Stack.

Answer

-

Status : -

Marks : 0/1

8. Consider a linked list implementation of stack data structure with three

operations:

push(value): Pushes an element value onto the stack.
pop(): Pops the top element from the stack.
top(): Returns the item stored at the top of the stack.

Given the following sequence of operations:

push(10);pop();push(5);top();

What will be the result of the stack after performing these operations?

Answer

-

Status : -

Marks : 0/1

9. The user performs the following operations on the stack of size 5 then at the end of the last operation, the total number of elements present in the stack is

push(1);
pop();
push(2);
push(3);
pop();
push(4);
pop();
pop();
push(5);

Answer

-

Status : -

Marks : 0/1

10. What will be the output of the following code?

```
#include <stdio.h>
#define MAX_SIZE 5
```

```

int stack[MAX_SIZE];
int top = -1;
int isEmpty() {
    return (top == -1);
}
int isFull() {
    return (top == MAX_SIZE - 1);
}
void push(int item) {
    if (isFull())
        printf("Stack Overflow\n");
    else
        stack[++top] = item;
}
int main() {
    printf("%d\n", isEmpty());
    push(10);
    push(20);
    push(30);
    printf("%d\n", isFull());
    return 0;
}

```

Answer

-

Status : -

Marks : 0/1

11. In an array-based stack, which of the following operations can result in a Stack underflow?

Answer

-

Status : -

Marks : 0/1

12. Here is an Infix Expression: $4+3*(6*3-12)$. Convert the expression from Infix to Postfix notation. The maximum number of symbols that will appear

on the stack AT ONE TIME during the conversion of this expression?

Answer

-

Status : -

Marks : 0/1

13. What is the value of the postfix expression 6 3 2 4 + - *?

Answer

-

Status : -

Marks : 0/1

14. In the linked list implementation of the stack, which of the following operations removes an element from the top?

Answer

-

Status : -

Marks : 0/1

15. Consider the linked list implementation of a stack.
Which of the following nodes is considered as Top of the stack?

Answer

-

Status : -

Marks : 0/1

16. The result after evaluating the postfix expression 10 5 + 60 6 / * 8 - is

Answer

-

Status : -

Marks : 0/1

17. Pushing an element into the stack already has five elements. The stack size is 5, then the stack becomes

Answer

-

Status : -

Marks : 0/1

18. In a stack data structure, what is the fundamental rule that is followed for performing operations?

Answer

-

Status : -

Marks : 0/1

19. What is the advantage of using a linked list over an array for implementing a stack?

Answer

-

Status : -

Marks : 0/1

20. What is the primary advantage of using an array-based stack with a fixed size?

Answer

-

Status : -

Marks : 0/1

Rajalakshmi Engineering College

Name: vasugi e.v.n
Email: 240801370@rajalakshmi.edu.in
Roll no: 240801370
Phone: 7708989508
Branch: REC
Department: I ECE AF
Batch: 2028
Degree: B.E - ECE

Scan to verify results



NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 4_MCQ_Updated

Attempt : 1
Total Mark : 20
Marks Obtained : 17

Section 1 : MCQ

1. The essential condition that is checked before insertion in a queue is?

Answer

Overflow

Status : Correct

Marks : 1/1

2. When new data has to be inserted into a stack or queue, but there is no available space. This is known as

Answer

overflow

Status : Correct

Marks : 1/1

3. In a linked list implementation of a queue, front and rear pointers are tracked. Which of these pointers will change during an insertion into a non-empty queue?

Answer

Only front pointer

Status : Wrong

Marks : 0/1

4. What is the functionality of the following piece of code?

```
public void function(Object item)
{
    Node temp=new Node(item,trail);
    if(isEmpty())
    {
        head.setNext(temp);
        temp.setNext(trail);
    }
    else
    {
        Node cur=head.getNext();
        while(cur.getNext()!=trail)
        {
            cur=cur.getNext();
        }
        cur.setNext(temp);
    }
    size++;
}
```

Answer

Insert at the rear end of the dequeue

Status : Correct

Marks : 1/1

5. After performing this set of operations, what does the final list look to contain?

```
InsertFront(10);
InsertFront(20);
InsertRear(30);
DeleteFront();
InsertRear(40);
InsertRear(10);
DeleteRear();
InsertRear(15);
display();
```

Answer

10 30 40 15

Status : Correct

Marks : 1/1

6. In linked list implementation of a queue, the important condition for a queue to be empty is?

Answer

FRONT is null

Status : Correct

Marks : 1/1

7. Which operations are performed when deleting an element from an array-based queue?

Answer

Dequeue

Status : Correct

Marks : 1/1

8. A normal queue, if implemented using an array of size MAX_SIZE, gets full when

Answer

Front = (rear + 1)mod MAX_SIZE

Status : Wrong

Marks : 0/1

9. What does the front pointer in a linked list implementation of a queue contain?

Answer

The address of the first element

Status : Correct

Marks : 1/1

10. In what order will they be removed If the elements "A", "B", "C" and "D" are placed in a queue and are deleted one at a time

Answer

ABCD

Status : Correct

Marks : 1/1

11. Front and rear pointers are tracked in the linked list implementation of a queue. Which of these pointers will change during an insertion into the EMPTY queue?

Answer

Both front and rear pointer

Status : Correct

Marks : 1/1

12. What will be the output of the following code?

```
#include <stdio.h>
#include <stdlib.h>
#define MAX_SIZE 5
typedef struct {
    int* arr;
    int front;
    int rear;
    int size;
} Queue;
Queue* createQueue() {
```

```

Queue* queue = (Queue*)malloc(sizeof(Queue));
queue->arr = (int*)malloc(MAX_SIZE * sizeof(int));
queue->front = -1;
queue->rear = -1;
queue->size = 0;
return queue;
}
int isEmpty(Queue* queue) {
    return (queue->size == 0);
}
int main() {
    Queue* queue = createQueue();
    printf("Is the queue empty? %d", isEmpty(queue));
    return 0;
}

```

Answer

Is the queue empty? 1

Status : Correct

Marks : 1/1

13. Which of the following can be used to delete an element from the front end of the queue?

Answer

```

public Object deleteFront() throws emptyDEQException{if(isEmpty())throw new
emptyDEQException("Empty");else{Node temp = head.getNext();Node cur =
temp;Object e = temp.getEle();head.setNext(cur);size--;return e;}}

```

Status : Wrong

Marks : 0/1

14. Which one of the following is an application of Queue Data Structure?

Answer

All of the mentioned options

Status : Correct

Marks : 1/1

15. The process of accessing data stored in a serial access memory is similar to manipulating data on a

Answer

Queue

Status : Correct

Marks : 1/1

16. Which of the following properties is associated with a queue?

Answer

First In First Out

Status : Correct

Marks : 1/1

17. What will be the output of the following code?

```
#include <stdio.h>
#define MAX_SIZE 5
typedef struct {
    int arr[MAX_SIZE];
    int front;
    int rear;
    int size;
} Queue;

void enqueue(Queue* queue, int data) {
    if (queue->size == MAX_SIZE) {
        return;
    }
    queue->rear = (queue->rear + 1) % MAX_SIZE;
    queue->arr[queue->rear] = data;
    queue->size++;
}

int dequeue(Queue* queue) {
    if (queue->size == 0) {
        return -1;
    }
}
```

```

    int data = queue->arr[queue->front];
    queue->front = (queue->front + 1) % MAX_SIZE;
    queue->size--;
    return data;
}
int main() {
    Queue queue;
    queue.front = 0;
    queue.rear = -1;
    queue.size = 0;
    enqueue(&queue, 1);
    enqueue(&queue, 2);
    enqueue(&queue, 3);
    printf("%d ", dequeue(&queue));
    printf("%d ", dequeue(&queue));
    enqueue(&queue, 4);
    enqueue(&queue, 5);
    printf("%d ", dequeue(&queue));
    printf("%d ", dequeue(&queue));
    return 0;
}

```

Answer

1 2 3 4

Status : Correct

Marks : 1/1

18. Insertion and deletion operation in the queue is known as

Answer

Enqueue and Dequeue

Status : Correct

Marks : 1/1

19. What will the output of the following code?

```

#include <stdio.h>
#include <stdlib.h>

```

```
typedef struct {
    int* arr;
    int front;
    int rear;
    int size;
} Queue;
Queue* createQueue() {
    Queue* queue = (Queue*)malloc(sizeof(Queue));
    queue->arr = (int*)malloc(5 * sizeof(int));
    queue->front = 0;
    queue->rear = -1;
    queue->size = 0;
    return queue;
}
int main() {
    Queue* queue = createQueue();
    printf("%d", queue->size);
    return 0;
}
```

Answer

0

Status : Correct

Marks : 1/1

20. What are the applications of dequeue?

Answer

All the mentioned options

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: vasugi e.v.n
Email: 240801370@rajalakshmi.edu.in
Roll no: 240801370
Phone: 7708989508
Branch: REC
Department: I ECE AF
Batch: 2028
Degree: B.E - ECE

Scan to verify results



NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 5_MCQ

Attempt : 1
Total Mark : 15
Marks Obtained : 13

Section 1 : MCQ

1. The preorder traversal of a binary search tree is 15, 10, 12, 11, 20, 18, 16, 19. Which one of the following is the postorder traversal of the tree?

Answer

11, 12, 10, 16, 19, 18, 20, 15

Status : Correct

Marks : 1/1

2. Which of the following is the correct post-order traversal of a binary search tree with nodes: 50, 30, 20, 55, 32, 52, 57?

Answer

50, 20, 30, 52, 57, 55, 32

Status : Wrong

Marks : 0/1

3. Find the in-order traversal of the given binary search tree.

Answer

1, 2, 4, 13, 14, 18

Status : Correct

Marks : 1/1

4. Which of the following is a valid preorder traversal of the binary search tree with nodes: 18, 28, 12, 11, 16, 14, 17?

Answer

18, 12, 11, 16, 14, 17, 28

Status : Correct

Marks : 1/1

5. Which of the following is the correct pre-order traversal of a binary search tree with nodes: 50, 30, 20, 55, 32, 52, 57?

Answer

50, 30, 20, 32, 55, 52, 57

Status : Correct

Marks : 1/1

6. Which of the following is the correct in-order traversal of a binary search tree with nodes: 9, 3, 5, 11, 8, 4, 2?

Answer

2, 3, 4, 5, 8, 9, 11

Status : Correct

Marks : 1/1

7. Find the postorder traversal of the given binary search tree.

Answer

13, 2, 1, 4, 14, 18

Status : Wrong

Marks : 0/1

8. Find the pre-order traversal of the given binary search tree.

Answer

13, 2, 1, 4, 14, 18

Status : Correct

Marks : 1/1

9. While inserting the elements 5, 4, 2, 8, 7, 10, 12 in a binary search tree, the element at the lowest level is _____.

Answer

12

Status : Correct

Marks : 1/1

10. Find the preorder traversal of the given binary search tree.

Answer

9, 2, 1, 6, 4, 7, 10, 14

Status : Correct

Marks : 1/1

11. Find the post-order traversal of the given binary search tree.

Answer

10, 17, 20, 18, 15, 32, 21

Status : Correct

Marks : 1/1

12. While inserting the elements 71, 65, 84, 69, 67, 83 in an empty binary search tree (BST) in the sequence shown, the element in the lowest level is _____.

Answer

67

Status : Correct

Marks : 1/1

13. In a binary search tree with nodes 18, 28, 12, 11, 16, 14, 17, what is the value of the left child of the node 16?

Answer

14

Status : Correct

Marks : 1/1

14. How many distinct binary search trees can be created out of 4 distinct keys?

Answer

14

Status : Correct

Marks : 1/1

15. Which of the following operations can be used to traverse a Binary Search Tree (BST) in ascending order?

Answer

Inorder traversal

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: vasugi e.v.n
Email: 240801370@rajalakshmi.edu.in
Roll no: 240801370
Phone: 7708989508
Branch: REC
Department: I ECE AF
Batch: 2028
Degree: B.E - ECE

Scan to verify results



NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 6_MCQ_Updated_1

Attempt : 1
Total Mark : 20
Marks Obtained : 19

Section 1 : MCQ

1. In a quick sort algorithm, where are smaller elements placed to the pivot during the partition process, assuming we are sorting in increasing order?

Answer

To the left of the pivot

Status : Correct

Marks : 1/1

2. What is the main advantage of Quicksort over Merge Sort?

Answer

Quicksort requires less auxiliary space

Status : Correct

Marks : 1/1

3. Consider the Quick Sort algorithm, which sorts elements in ascending order using the first element as a pivot. Then which of the following input sequences will require the maximum number of comparisons when this algorithm is applied to it?

Answer

22 25 56 67 89

Status : Correct

Marks : 1/1

4. What happens when Merge Sort is applied to a single-element array?

Answer

The array remains unchanged and no merging is required

Status : Correct

Marks : 1/1

5. Which of the following strategies is used to improve the efficiency of Quicksort in practical implementations?

Answer

Choosing the pivot randomly or using the median-of-three method

Status : Correct

Marks : 1/1

6. Which of the following statements is true about the merge sort algorithm?

Answer

It requires additional memory for merging

Status : Correct

Marks : 1/1

7. Merge sort is _____.

Answer

Comparison-based sorting algorithm

Status : Correct

Marks : 1/1

8. Which of the following modifications can help Quicksort perform better on small subarrays?

Answer

Switching to Insertion Sort for small subarrays

Status : Correct

Marks : 1/1

9. Is Merge Sort a stable sorting algorithm?

Answer

Yes, always stable.

Status : Correct

Marks : 1/1

10. Which of the following sorting algorithms is based on the divide and conquer method?

Answer

Merge Sort

Status : Correct

Marks : 1/1

11. What is the best sorting algorithm to use for the elements in an array that are more than 1 million in general?

Answer

Quick sort.

Status : Correct

Marks : 1/1

12. Which of the following scenarios is Merge Sort preferred over Quick Sort?

Answer

When sorting linked lists

Status : Correct

Marks : 1/1

13. Which of the following methods is used for sorting in merge sort?

Answer

merging

Status : Correct

Marks : 1/1

14. What happens during the merge step in Merge Sort?

Answer

Two sorted subarrays are combined into one sorted array

Status : Correct

Marks : 1/1

15. In a quick sort algorithm, what role does the pivot element play?

Answer

It is used to partition the array

Status : Correct

Marks : 1/1

16. Let P be a quick sort program to sort numbers in ascending order using the first element as a pivot. Let t_1 and t_2 be the number of comparisons made by P for the inputs $\{1, 2, 3, 4, 5\}$ and $\{4, 1, 5, 3, 2\}$, respectively. Which one of the following holds?

Answer

$t_1 > t_2$

Status : Correct

Marks : 1/1

17. Which of the following is true about Quicksort?

Answer

It is an in-place sorting algorithm

Status : Correct

Marks : 1/1

18. Why is Merge Sort preferred for sorting large datasets compared to Quick Sort?

Answer

Merge Sort has better worst-case time complexity

Status : Correct

Marks : 1/1

19. The following code snippet is an example of a quick sort. What do the 'low' and 'high' parameters represent in this code?

```
void quickSort(int arr[], int low, int high) {  
    if (low < high) {  
        int pivot = partition(arr, low, high);  
        quickSort(arr, low, pivot - 1);  
        quickSort(arr, pivot + 1, high);  
    }  
}
```

Answer

The range of elements to sort within the array

Status : Correct

Marks : 1/1

20. Which of the following is not true about QuickSort?

Answer

It as an adaptive sorting algorithm

Status : Wrong

Marks : 0/1

Rajalakshmi Engineering College

Name: vasugi e.v.n
Email: 240801370@rajalakshmi.edu.in
Roll no: 240801370
Phone: 7708989508
Branch: REC
Department: I ECE AF
Batch: 2028
Degree: B.E - ECE

Scan to verify results



NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 7_MCQ_Updated

Attempt : 1
Total Mark : 20
Marks Obtained : 19

Section 1 : MCQ

1. Which of these hashing methods may result in more uniform distribution with small keys?

Answer

Mid-Square

Status : Correct

Marks : 1/1

2. Which folding method divides the key into equal parts, reverses some of them, and then adds all parts?

Answer

Folding reversal method

Status : Correct

Marks : 1/1

3. In the division method of hashing, the hash function is typically written as:

Answer

$$h(k) = k \% m$$

Status : Correct

Marks : 1/1

4. What is the primary disadvantage of linear probing?

Answer

Clustering

Status : Correct

Marks : 1/1

5. Which situation causes clustering in linear probing?

Answer

All the mentioned options

Status : Correct

Marks : 1/1

6. In the folding method, what is the primary reason for reversing alternate parts before addition?

Answer

To reduce the chance of collisions caused by similar digit patterns

Status : Correct

Marks : 1/1

7. What happens if we do not use modular arithmetic in linear probing?

Answer

Index goes out of bounds

Status : Correct

Marks : 1/1

8. In C, how do you calculate the mid-square hash index for a key k, assuming we extract two middle digits and the table size is 100?

Answer

$((k * k) / 10) \% 100$

Status : Wrong

Marks : 0/1

9. In linear probing, if a collision occurs at index i, what is the next index checked?

Answer

$(i + 1) \% \text{table_size}$

Status : Correct

Marks : 1/1

10. What is the initial position for a key k in a linear probing hash table?

Answer

$k \% \text{table_size}$

Status : Correct

Marks : 1/1

11. What is the worst-case time complexity for inserting an element in a hash table with linear probing?

Answer

$O(n)$

Status : Correct

Marks : 1/1

12. In division method, if key = 125 and m = 13, what is the hash index?

Answer

8

Status : Correct

Marks : 1/1

13. Which C statement is correct for finding the next index in linear probing?

Answer

$\text{index} = (\text{index} + 1) \% \text{size};$

Status : Correct

Marks : 1/1

14. What would be the result of folding 123456 into three parts and summing: $(12 + 34 + 56)$?

Answer

102

Status : Correct

Marks : 1/1

15. Which of the following best describes linear probing in hashing?

Answer

Resolving collisions by linearly searching for the next free slot

Status : Correct

Marks : 1/1

16. What is the output of the mid-square method for a key $k = 123$ if the hash table size is 10 and you extract the middle two digits of $k * k$?

Answer

1

Status : Correct

Marks : 1/1

17. Which of the following values of 'm' is recommended for the division method in hashing?

Answer

A prime number

Status : Correct

Marks : 1/1

18. What does a deleted slot in linear probing typically contain?

Answer

A special "deleted" marker

Status : Correct

Marks : 1/1

19. Which of the following statements is TRUE regarding the folding method?

Answer

It divides the key into parts and adds them.

Status : Correct

Marks : 1/1

20. Which data structure is primarily used in linear probing?

Answer

Array

Status : Correct

Marks : 1/1