

Name:

Enrolment No:



UNIVERSITY OF PETROLEUM AND ENERGY STUDIES

End Semester Examination, July 2020

Course: Advanced Data Structures

Program: B.Tech. CS OSS

Course Code: CSEG 1004

Semester: II

Time 02 hrs.

Max. Marks: 100

Mode of exam: Online through blackboard

Question 1

1. Write down post order traversal (Use comma separation) of binary tree, given

Preorder traversal sequence: G,B,Q,A,C,K,F,P,D,E,R,H

In order traversal sequence: Q,B,K,C,F,A,G,P,E,D,H,R

4 points

Question 2

1. Balanced m-way search tree is called as B-tree.

True

False

1 points

Question 3

1. Select all correct answers.

A non empty B -tree of order m, is an m-way search tree in which

all leaf nodes are in the same level

The root has at least two child nodes and at most m child nodes.

- the number of keys in each internal node is one less than the number of child nodes and these keys partition the keys in the subtrees of the node in a manner similar to that of m-way search trees
- the internal nodes except the root have at least $\text{ceil}(m/2)$ child nodes and at most m child nodes

4 points

Question 4

1. What is simple uniform hashing?

- Every element has equal probability of hashing into any of the slots
- A weighted probabilistic method is used to hash elements into the slots
- Elements has Random probability of hashing into array slots
- Elements are hashed based on priority

1 points

Question 5

1. The case in which a key other than the desired one is kept at the identified location is called?

- Chaining
- Open addressing
- Collision
- Hashing

1 points

Question 6

1. What maximum difference in heights between the leafs of a AVL tree is possible?

- atmost 1
- $\log(n)$ where n is the number of nodes
- 0 or 1
- n where n is the number of nodes

1 points

Question 7

1. Which of the following is the hashing function for separate chaining
 - $H(x)=x \bmod (\text{table size} * 2)$
 - $H(x)=\text{hash}(x)+i^2 \bmod \text{table size}$
 - $H(x)=x \bmod \text{table size}$
 - $H(x)=(\text{hash}(x)+f(i)) \bmod \text{table size}$

3 points

Question 8

1. Which of the following is a disadvantage of using separate chaining using linked lists
 - It uses array
 - It does not resolve collision
 - It requires many pointers
 - It requires linked lists

1 points

Question 9

1. Which of the following technique stores data in a separate entity in case of a collision

- Double hashing
- Linear probing
- Open addressing
- Chaining using doubly linked list

1 points

Question 10

1. What is the advantage of using a doubly linked list for chaining over singly linked list?
 - it takes less memory
 - it is easy to implement
 - it makes the process of insertion and deletion faster
 - it causes less collisions

1 points

Question 11

1. What is the time complexity of insert function in a hash table using a binary tree
 - $O(1)$
 - $O(n)$
 - $O(\log n)$
 - $O(n \log n)$

1 points

Question 12

1. What is the advantage of a hash table over BST?

- "hash table has a better average time complexity for performing insert, delete and search operations"
- hash table requires less space
- range query is easy with hash table
- easier to implement

1 points

Question 13

1. What is the time complexity of delete function in the hash table using list head

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

1 points

Question 14

1. How many probes are required on average for insertion and successful search?

- 4 and 10
- 2 and 6
- 2.5 and 1.5
- 3.5 and 1.5

1 points

Question 15

1. Which of the following is the correct function definition for linear probing?

- $F(i)=i$
- $F(i)=i+1$
- $F(i)=i^2$
- $F(i)= 1$

1 points

Question 16

1. What is the hash function used in linear probing?

- $H(x)= \text{key mod table size}$
- $H(x)= (\text{key}+ F(i^2)) \text{ mod table size}$
- $H(x)= X \text{ mod } 17$
- $H(x)= (\text{key}+ F(i)) \text{ mod table size}$

1 points

Question 17

1. "In quadratic probing, if the table size is prime, a new element cannot be inserted if the table is half full."

- True
- False

1 points

Question 18

1. Which of the following techniques offer better cache performance?

- Quadratic probing

- Linear probing
- Double hashing
- Rehashing

1 points

Question 19

1. Which scheme uses a randomization approach?

- hashing by multiplication
- hashing by division
- universal hashing
- open addressing

1 points

Question 20

1. "Assuming value of every weight to be greater than 10, in which of the following cases the shortest path of a directed weighted graph from 2 vertices u and v will never change?"

- multiply all values by 10
- add all values by 10
- in both the cases of multiplying and adding by 10
- subtract 10 from all the values

1 points

Question 21

1. What is the maximum possible number of edges in a directed graph with no self loops having 8 vertices?

- 28
- 64
- 56
- 256

1 points

Question 22

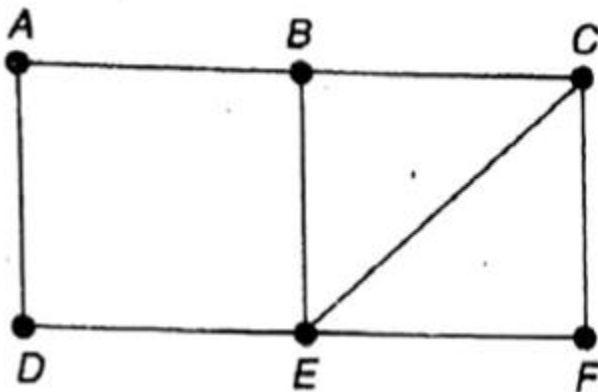
1. "With V (greater than 1) vertices, how many edges at most can a Directed Acyclic Graph possess?"

- $(V-1)C2$
- $(V+1)C2$
- $(V*(V-1))/2$
- $(V*(V+1))/2$

1 points

Question 23

1. Consider the connected graph:



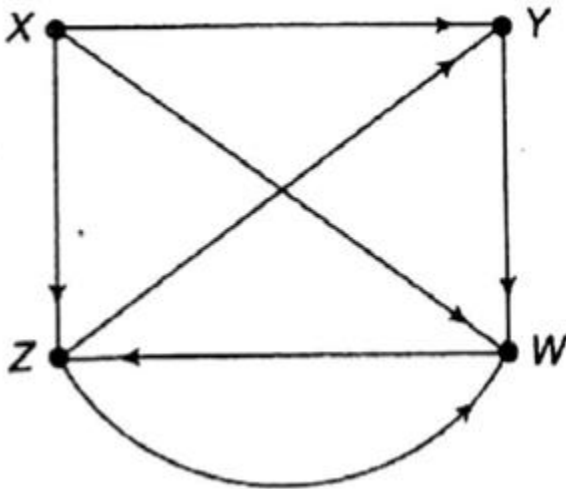
Calculate the distance between A and F

- 2
- 3
- 4
- No path from A to F

1 points

Question 24

1. Consider the given directed graph



calculate in -degree and out degree for node Y.

- 1, 2
- 2, 1
- 0, 2
- 1, 0

2 points

Question 25

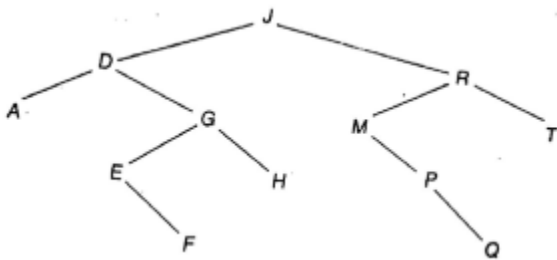
1. The min number of edges required to create a cyclid graph of n vertices is

- n
- 2n
- n-1
- n+1

1 points

Question 26

1. Consider the binary search tree T:



1. Perform following operations in it

1. delete node M
2. then delete node D

what will be the in order traversal of updated binary search tree

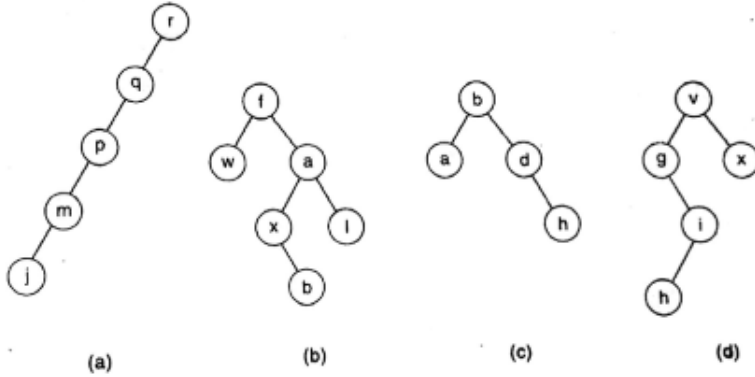
- A,E,F,G,H,J,P,Q,R,T
- J,E,R,A,G,P,T,F,H,Q
- A,E,F,H,G,P,Q,R,T,J
- A,E,F,G,H,P,Q,R,T,J

3 points

Question 27

1. Do the correct matching

Do the correct matching



(a)

(b)

(c)

(d)

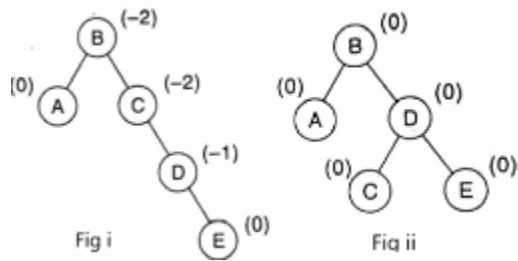
- A. Skewed binary search tree
- B. Binary tree
- C. AVL search tree
- D. Binary search tree

Question 28

1. Insert following letter into AVL search tree

A,B,C,D,E

The output will be:



1.

- fig i
- fig ii

- both fig i and ii are incorrect

2 points

Question 29

1. Consider for binary tree:

preorder: A,B,C,D,E,F,G,H,J,K,L,M,P,Q,N

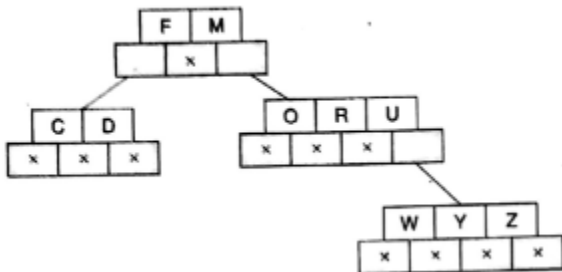
Inorder: C,D,E,B,G,H,F,K,L,P,Q,M,N,J,A

Write post order sequence

5 points

Question 30

1. In the following 4-way search tree, trace the tree after deletion of i. U and then ii. M . What will be the last leaf node.



1.

- C
- D
- Y Z
- Z
- W Y Z

5 points

Question 31

1. . can be invoked like a normal function without the help of any object.

- constant member function
- private member function
- static member function
- friend function

1 points

Question 32

1. .. allows memory dumping on a bit by bit basis from one object to another

- Shallow Copy
- Deep Copy
- Inheritance
- Copy constructor

1 points

Question 33

1. " allows us to group a set of global classes, objects and/or functions under a specific name."

- Storage Classes
- Global variable
- Namespace
- None of these

1 points

Question 34

1. "In nested try blocks, there is no need to specify catch handler for inner try block. Outer catch handler is sufficient for the program."

- True
- False

1 points

Question 35

1. "Insert the following data in an AVL tree and mention which rotation will be performed to balance the tree: 13,16,10,6,52,3 12"

- Left Rotation
- Right Rotation
- Left Right Rotation
- Right Left Rotation

1 points

Question 36

1. "To delete a dynamically allocated array named a , the correct statement is"

- delete a;
- delete []a;
- delete a[0];
- delete [0]a;

1 points

Question 37

1. "While declaring a static member in class, which statement is true?"

- a static member can be a public member
- a static member can be a private member
- a static member can be a protected member
- All of these

1 points

Question 38

1. "While using an object as a function argument, a copy of the entire object is passed to the function in .. method."

- pass-by-value
- pass-by-reference
- pass-by-variable
- pass-by-function

1 points

Question 39

1. A file in C++ can be opened using:

- constructor of the appropriate class
- open() function
- Both of these
- None of these

1 points

Question 40

1.

State whether the following statements are True or False about the characteristics of static data members.

i) Only one copy of a static member is created for the entire class and is shared by all the objects of that class, no matter how many objects are created.

ii) The static member variable is visible only within the class, but its lifetime is the entire program.

i-True, ii-True

i-False, ii-True

i-True, ii-False



i-True, ii-True

2 points

Question 41

1.

What Will be the output of the following program?

```
#include <iostream>

using namespace std;

class Test

{ public:

    int x;

    mutable int y;

    Test()

        { x = 4; y = 10; }

};

int main()

{ const Test t1;

    t1.x = 8;

    cout << t1.x;
```

```
t1.y = 20;  
cout << t1.y;  
return 0; }
```

x=4, y=10

x=8, y=20

x=4, y=20

compilation error

2 points

Question 42

1.

what is the output of following code?

```
#include <iostream>  
using namespace std;  
static int sum=50;
```

```
int main()
{
    for (int i = 0; i < 5; ++i)
        { sum = sum+i; }
    cout << sum << endl;

    return 0; }
```

- 50
- 10
- 60
-

Undefined value

2 points

Question 43

1.

What is the output of the following code?

```
#include <iostream>

using namespace std;

class Demo
{ public:
    int *ptr;

    int getLength( void );
```

```

Demo( int len );

Demo( const Demo &obj);

~Demo(); };

Demo::Demo(int len)

{ cout << "Normal constructor allocating ptr" << endl;

  ptr = new int;

  *ptr = len;

}

Demo::Demo(const Demo &obj)

{ cout << "Copy constructor allocating ptr." << endl;

  ptr = new int;

  *ptr = *obj.ptr;

}

Demo::~Demo(void)

{cout << "Freeing memory!" << endl;

  delete ptr;

}

int Demo::getLength( void )

{ return *ptr; }

void show(Demo obj)

{ cout << "Length of Demo : " << obj.getLength() <<endl; }

int main()

{ Demo demo1(10);

```

```
Demo demo2 = demo1;  
show(demo1);  
show(demo2);  
return 0; }
```



Normal constructor
allocating ptr

Copy constructor
allocating ptr.

Length of line : 10

Freeing memory!

Freeing memory!



Normal constructor
allocating ptr

Copy constructor
allocating ptr.

Length of line : 10

Freeing memory!



Normal constructor
allocating ptr

Copy constructor
allocating ptr.

Length of line : 10



Normal constructor
allocating ptr

Copy constructor
allocating ptr.

Copy constructor
allocating ptr.

Length of line : 10

Freeing memory!

Copy constructor
allocating ptr.

Length of line : 10

Freeing memory!

Freeing memory!

Freeing memory!

4 points

Question 44

1.

What is the output of the following code?

```
#include<iostream>
```

```
using namespace std;
```

```
{ int a = 6;
```

```
int &n = a;
```

```
n=a++;
```

```
a=n++;
```

```
cout<<endl;
```

}

6,8

7,6

6,7

None of these

2 points

Question 45

1.

What is the output of the following code?

```
#include<iostream>
```

```
using namespace std;
```

```
int main()
```

```
{ int a=9,x;
```

```
  x = ++a * --a;
```

```
  cout<<++a<< " " << a++ << " " << x <<endl;
```


}

10, 8, 100

11, 10, 90

11, 9, 81

11, 11, 100

2 points

Question 46

```
1. #include<iostream>
using namespace std;
int main()
{
char s1[]="HELLO STUDENTS" , s2[]="HELLO STUDETNS";
int Z=(s1==s2);
if(Z)
cout<<Z;
```

```
else  
cout<<Z;  
}
```

- 0
- 1
- Error
- None

2 points

Question 47

1.

what is the output of following code?

```
#include<iostream>  
  
using namespace std;  
  
int hello(int a, int b=2)  
{int r;  
  
r = a*b;  
  
return r;  
  
}  
  
int main()  
{ Cout<<hello (6) <<" , "<<hello (2, 3) ;  
  
}
```

- 6, 12
- 12, 6
- 2, 3
- 6, 3

2 points

Question 48

1.

What will be the output of the following program?

```
#include<iostream>

using namespace std;

class student

{ public :

    int marks;

    void disp()

        { cout<<"its base class";}

};

class topper:public student

{public :

    void disp()

        { cout<<"Its derived class"; }

};

int main()
```

```
{ student s;  
  topper t;  
  s.disp();  
  t.disp();  
}
```

Its base classIts base
class

Its base classIts derived
class

Its derived classIts
base class

Its derived classIts
derived class

3 points

Question 49

1.

What will be the output of the following program?

```
#include<iostream>

using namespace std;

int main()
{
    try
        {throw 5;}
    catch (...)
        {cout << "Default Exception";}
    catch (int arg)
        {cout << "Int Exception";}
    return 0;}

```



Default Exception



Int Exception



Compiler Error



None of the above

2 points

Question 50

1.

What will be the output of the following program?

```
#include <iostream>

using namespace std;

class base
{
public:
    virtual void show()=0;
    void display()
        {cout<<"Base class"<<endl;}
};

class derived:public base
{public:
    void display()
        {cout << "Derived class" <<endl;}
};

int main()
    { derived obj;
```

```
obj.display();  
return 0;  
}
```

Derived class

Base class

Compile time error

None of these

3 points

Question 51

1. "Which of the following statement(s) is/are true about operator overloading?"

(A) In Unary Operator Overloading, one argument is passed to the operator function.

(B) In Binary operator overloading two arguments are passed to the operator function."

- Only (A)
- Only (B)
- Both (A) and (B)
- None of these

1 points

Question 52

1. Which among the following is the correct syntax for multiple inheritance?

- ```
"class student
{public:
 int marks;
}s;
class stream
{int total;};
class topper:public student, public stream{ };"
```
- ```
class student
{int marks;};
class stream{ };
class topper: public student{ };
class student
{int marks;};
class stream:public student{ };
```


- class student{ };
- class stream{ };
- class topper{ };

1 points

Question 53

1. Which of the following statement(s) is/are true about templates?

(A) Template is a feature of C++ that allows us to write one code for different data types.

(B) We can write one function that can be used for all data types including user defined types.

(C) We can create one class or structure that can be used for all data types including user defined types.

(D) Template is an example of compile time polymorphism.

- (A) and (D)
- "(A) , (B) and (D)"
- "(A), (B), and (C)"
- All of these

1 points

Question 54

1. Encapsulation is ..

- The process of binding together the data and functions in a class.
- The process of hiding the internal implementations and displaying only the required details.
- The process of creating and initializing the default constructor automatically
- The process of using the private and protected members outside the class.

1 points

Question 55

1. Which among the following best defines static variables members?
 - Data which is allocated for each object separately
 - Data which is common to all the objects of a class
 - Data which is common to all the classes
 - Data which is common to a specific method

1 points

Question 56

1. Which among the following best describes the Inheritance?
 - Copying the code already written
 - Using the code already written once
 - Using already defined functions in programming language
 - Using the data and functions into derived segment

1 points

Question 57

1. Which among the following is wrong syntax related to static data members?
 - `className : dataType -> memberName;`
 - `className :: staticDataMember;`
 - `dataType className :: memberName =value;`
 - `static dataType memberName;`

1 points

Question 58

1. Which of the following is not a file mode in C++ ?

- ios::ate
- ios::octal
- ios::binary
- ios::nocreate

1 points

Question 59

1. which statement is true in case of constructor?

- constructor must be declared in public part otherwise result will be error.
- Constructor cannot be virtual.
- "They do not have return types, not even void"
- All of these

1 points

Question 60

1.

What is the output of the following program?

```
#include<iostream>
using namespace std;
int main()
```

```
{
    char s[] = "Fine";
    *s = 'N';
    cout<<s<<endl;
}
```

- Fine
- Nine
- Runtime Error
- Compile time Error

3 points