Hall Ticket Number:

Code No. : 12015 DS

VASAVI COLLEGE OF ENGINEERING *(Autonomous)*, HYDERABAD

B.E. (CBCS) II-Semester New Examinations, May/June-2018

Data Structures

(IT)

Time: **3** hours Max. Marks: **60**

*Note: Answer **ALL** questions in **Part-A** and any **FIVE** from **Part-B***

Part-A (10 × 2 = 20 Marks)

1. What is the output of this program?

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
int *ptr;
*ptr = 10;
*ptr = 20;
printf("%d\n",*ptr);
return 0;
}
```

2. What is the output of following program?

```
#include <stdio.h>
void fun(int x)
{
        x = 30;
}
int main()
{
        int y = 20;
        fun(y);
        printf("%d", y);
return 0;
}
```

3. Define ADT with an example.
4. Write the result of evaluating this postfix expression 5, 4, 6, +, *, 4, 9, 3, /, +, *
5. What is the maximum path length from start to finish in any maze of dimensions *n* X *m*?
6. How can you implement a queue with stack?
7. Write the pseudo code to add a node at the end of singly linked list, if the pointer is initially pointing to the head of the list?
8. Compare Double Linked List with Singly Linked List?
9. Sort the characters (using dictionary ordering relation A<B<C<.....<Z) in the word EXAMPLE using merge sort.
10. Write a piece of code to find an element using binary search technique.

Contd...2

Code No. : 12015 DS

Part-B (5 × 8 = 40 Marks)

11. a) Discuss how you would actually represent the list of name and telephone number pairs in a real machine. How would you handle people with the same last name? [4]

    b) Describe the various functions for the dynamic memory allocation and release [4]

12. a) Write a C program for converting an infix expression in to postfix expression. [6]

    b) Write the postfix form of the following expressions: [2]
    i) A ** B ** C ii) $-A + B - C + D$ iii) A ** $-B + C$ iv) $(A + B) * D + E/(F + A * D) + C$

13. a) Write insert and delete procedures of a circular queue being empty or full. [4]

    b) Write a C program to implement stack using arrays [4]

14. a) Write an algorithm LENGTH(P) to count the number of nodes in a singly linked list P, where P points to the first node in the list. The last node has link field 0. [4]

    b) Give an algorithm for a singly linked list which reverses the direction of the links. [4]

15. a) Explain the code snippet which performs unordered linear search iteratively? [3]

```
int unordered Linear Search(int arr[], int size, int data)
{
int index;
for(int i=0; i < size; i++)
{
if(arr[i]==data)
{
      index=i;
         break;
}
}
return index;
}
```

    b) Write an algorithm for partition exchange sort. Show the contents of the following after placing the pivot 47 at proper place 47, 21, 23, 56, 12, 87, 19, 35, 11, 36, 72, 12. [5]

16. a) Write a c program to find the smallest and second smallest element of an array without using any sorting technique. [4]

    b) Following is C like pseudo code of a function that takes a number as an argument and uses a stack S to do processing. [4]

```
void fun(int n)
{
Stack S;  // Say it creates an empty stack S
   while (n > 0)
{
   // This line pushes the value of n%2 to stack S
   push(&S, n%2);
   n = n/2;
}
// Run while Stack S is not empty
while (!isEmpty(&S))
 printf("%d ", pop(&S)); // pop an element from S and print it
}
```

What does the above function do in general?

Contd... 3

Code No. : 12015 DS

17. Answer any ***two*** of the following:

a) Following is C like pseudo code of a function that takes a Queue as an argument and uses a stack S to do processing. [4]

```
void fun(Queue *Q)
{
   Stack S;  // Say it creates an empty stack S
   // Run while Q is not empty
   while (!isEmpty(Q))
   {
// deQueue an item from Q and push the dequeued item to S
push(&S, deQueue(Q));
   }
// Run while Stack S is not empty
while (!isEmpty(&S))
   {
// Pop an item from S and enqueue the poppped item to Q
enQueue(Q, pop(&S));
   }
}
```

What does the above function do in general?

b) What is the functionality of the following code? [4]

```
public void function (Node node)
{
   if(size==0)
   head=node;
   else
   {
   Node temp,cur;
   for(cur=head;(temp=cur.getNext())!=null;cur=temp);
   cur.setNext(node);
   }
size++
}
```

c) Write an algorithm for insertion sort. Sort the sequence 3, 1, 4, 1, 5, 9, 2, 6, 5 using insertion sort by tabulating the values in each step. [4]