# Assignment-2

Institute/Department: AIT/CSE
Division: 6 th
Subject Name: Technical Training
Subject Code: CSY-355
Max. Marks: 12

1. What is a dequeue?
a) A queue with insert/delete defined for both front and rear ends of the queue
b) A queue implemented with a doubly linked list
c) A queue implemented with both singly and doubly linked lists
d) A queue with insert/delete defined for front side of the queue

2. Select the function which performs insertion at the front end of the dequeue?
a)

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

b)

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

c)

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

d)

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

3. 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++;
}
```

a) Insert at the front end of the dequeue
b) Insert at the rear end of the dequeue
c) Fetch the element at the rear end of the dequeue
d) Fetch the element at the front end of the dequeue

4. Which one of the following is an application of Stack Data Structure?
a. Managing function calls
b. The stock span problem
c. Arithmetic expression evaluation
d. All of the above

5. Which of the following is true about linked list implementation of stack?
a. In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end.
b. In push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from the beginning.
c. Both of the above
d. None of the above

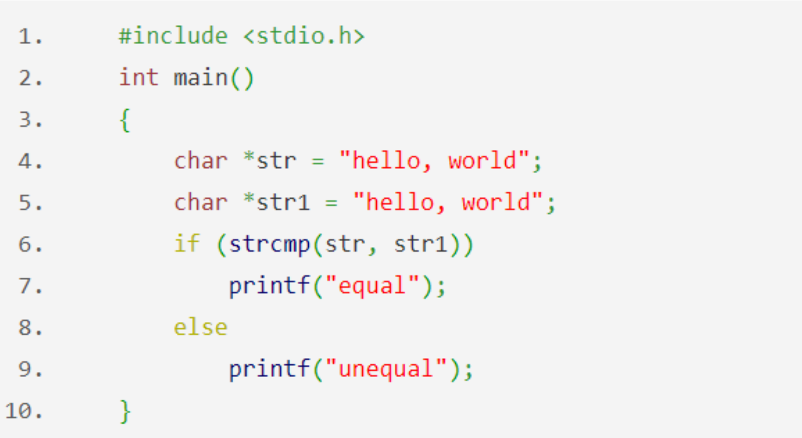
6. Pushing an element into stack already having five elements and stack size of 5, then stack becomes
a) Overflow
b) Crash
c) Underflow
d) User flow

7.

```
declare a stack of characters
while ( there are more characters in the word to read )
{
  read a character
  push the character on the stack
}
while ( the stack is not empty )
{
  pop a character off the stack
  write the character to the screen
}
```

What is output for input "geeksquiz"?
a. geeksquizgeeksquiz
b. ziuqskeeg
c. geeksquiz
d. ziuqskeegziuqskeeg

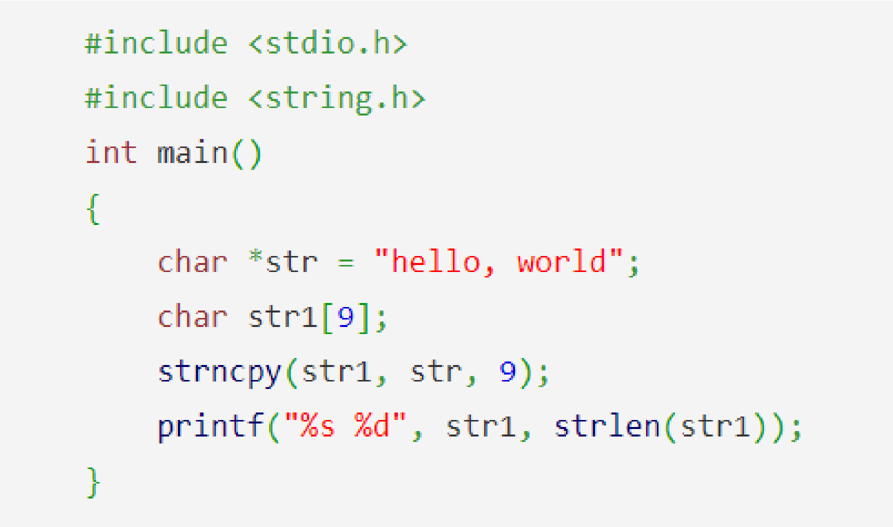
8. What will be the output of the following C code?



```
#include <stdio.h>
int main()
{
    char *str = "hello, world";
    char *str1 = "hello, world";
    if (strcmp(str, str1))
        printf("equal");
    else
        printf("unequal");
}
```

a.  equal
b.  unequal
c. Compilation error
d.  Depends on the compiler

9. What will be the output of the following C code?

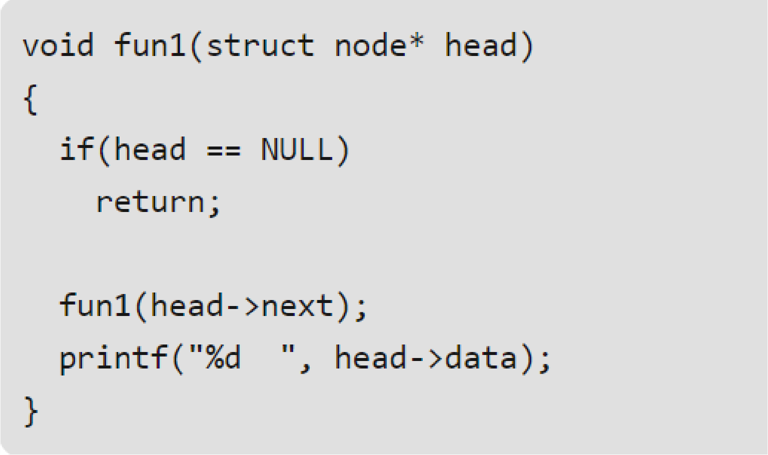


```
#include <stdio.h>
#include <string.h>
int main()
{
    char *str = "hello, world";
    char str1[9];
    strncpy(str1, str, 9);
    printf("%s %d", str1, strlen(str1));
}
```

a. Compilation error
b. Segmentation Fault
c. hello, world
d.  hello, wo 9

10. What will be the output of the following C code?
a. Compilation error
b. Undefined behaviour
c. 13
d. 11

11. What does the following function do for a given Linked List with first node as *head*?



```
void fun1(struct node* head)
{
  if(head == NULL)
    return;

  fun1(head->next);
  printf("%d  ", head->data);
}
```

a. Prints all nodes of linked lists
b. Prints all nodes of linked list in reverse order
c. Prints alternate nodes of Linked List
d. Prints alternate nodes in reverse order

12. Which of the following sorting algorithms can be used to sort a random linked list with minimum time complexity?
a. Insertion Sort
b. Quick Sort
c. Heap Sort
d. Merge Sort