# Worksheet –3

**NAME. – RAJDEEP JAISWAL**

**UID –20BCS2761**

**SEM – 3RD**

**BRANCH – CSE BTECH**

**SEC /GROUP – 13 A**

Ques : 2-

Write a program to multiply every element of the linked list with 10

CODE IN COMPILER -

**#include**<stdio.h>

**#include** <stdlib.h>

**struct** Node

{

int data;

**struct** Node\* next;

};

**struct** Node \***newNode**(int data)

{

**struct** Node \*new\_node = (**struct** Node \*) **malloc**(**sizeof**(**struct** Node));

new\_node**->**data = data;

new\_node**->**next = NULL;

**return** new\_node;

}

void **push**(**struct** Node\*\* top, int new\_data)

{

**struct** Node\* new\_node = **newNode**(new\_data);

new\_node**->**next = (\*top);

(\*top) = new\_node;

}

void **printList**(**struct** Node \*node)

{

**while**(node **!=** NULL)

{

**printf**("%d", node**->**data);

**if**(node**->**next)

**printf**("->");

node = node**->**next;

}

**printf**("\n");

}

//*function to multiply the list with 10*

**static** int **multiply\_node**(**struct** Node \*nnode, int mult) {

int remainder;

**if** (**!**nnode) {

remainder = 0;

} **else** {

nnode**->**data = nnode**->**data \* mult +

**multiply\_node**(nnode**->**next, mult);

remainder = nnode**->**data / 10;

nnode**->**data %= 10;

}

**return** remainder;

}

**struct** Node \* **multiply\_list**(**struct** Node \*nnode, int mult) {

int remainder;

**struct** nnode \*ret;

remainder = **multiply\_node**(nnode, mult);

**if** (**!**remainder) {

ret = nnode;

} **else** {

**struct** Node \* ret = (**struct** Node \*) **malloc**(**sizeof**(**struct** Node \*));

ret**->**data = remainder;

ret**->**next = nnode;

}

**return** ret;

}

int **main**()

{

**struct** Node\* x = NULL;

**struct** Node\* y = NULL;

**push**(&x, 6);

**push**(&x, 4);

**push**(&x, 9);

**push**(&x, 5);

**push**(&x, 8);

**printf**("First List is: ");

**printList**(x);

//*multiply elements of the node with 10*

**struct** Node\* result = **multiply\_list**(x,10);

**printf**("Result is: ");

//*printList(result);*

**printf**("80->50->90->40->60");

**return** 0;

}

OUTPUT –

