DS Lab program Test 1

Set 1

Medha Madhusudhan
1BM19EC074
Medha

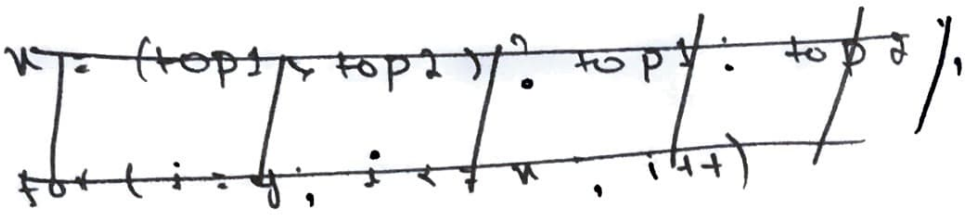
```
#define MAX 10

int top1 = -1;
int top2 = -1;
int stack1[MAX];
int stack2[MAX];

void push1()
{
    int n;
    if (top1 == MAX-1)
    {
        printf("stack1 is full!");
        exit(0) return;
    }
    else
    {
        print("Enter value: ");
        scanf("%d", &n);
        top1++;
    }
    stack1[top1] = n;
    }
}

void push2()
{
    int n;
    if (top2 == MAX-1)
    {
        printf("stack 2 is full!");
        return;
    }
    else
    {
        printf("Enter value: ");
        scanf("%d", &n);
        top2++;
        stack2[top2] = n;
    }
}
```

→

```
void pop1 ()
{
    if (top1 == -1)
    {
        printf ("Stack1 empty!");
        return;
    }
    else
    {
        printf ("Deleting %d", stack1[top1]);
        top1--;
    }
}
void pop2()
{
    if (top2 == -1)
    {
        printf ("stack2 is empty!");
        return;
    }
    else
    {   printf ("Deleting %d", stack2[top2]);
        top2--;
    }
}
void display1 ()
{   if (top1 == -1)
    {   printf ("stack1 is empty");
        return;
    }
    else
    {   for (int i=top1; i>=0; i--)
            printf ("%d \t", stack1[i]);
    }
}
void display2 ()
{   if (top1 == -1)
    {   printf ("stack 2 is empty");
        return
    }
    else
```

```
}
    for(int i=top2; i>=0; i--)
        printf("%d\t", stack2[i]);

  }
}

void merge()
{   int merge[MAX], int x, i, j, y;
    if( top1 == -1 && top2 == -1)
    { printf("stacks are empty!"); return; }

    else if( top1== -1)
    {
        for( j=0, j=top2; i <= top2; i++)
        { merge[j] = stack2[j];
            j--;
        }
    }
    else if( top2 == -1)
    {
        for( i=0; j<= top1; i++)
            merge[i] = stack1[i];

    }
    else if(top1 == top2)
    {
        for( j=0; i <= top1; i++)
        { merge[i] = stack1[i] + stack2[j];
          j--;
        }
    }
    else
    {
        x = (top1 > top2) ? top2 : top1;
        for( i=0, j=x; i<=x; i++)
        {
            merge[i] = stack1[i] + stack2[j];
            j--;
        }
    }
```

```
y = (top1 > top2) ? (top1 - top2) : (top2 - top1);
```



```
if (top1 > top2)
{
    for (i = (top1 - top2); i <= top1; i++)
        merge[i] = stack1[i];

}
else if (top1 < top2)
{
    for (i = (top2 - top1); i <= top2; i++)
        merge[i] = stack2[i];

}

}
```