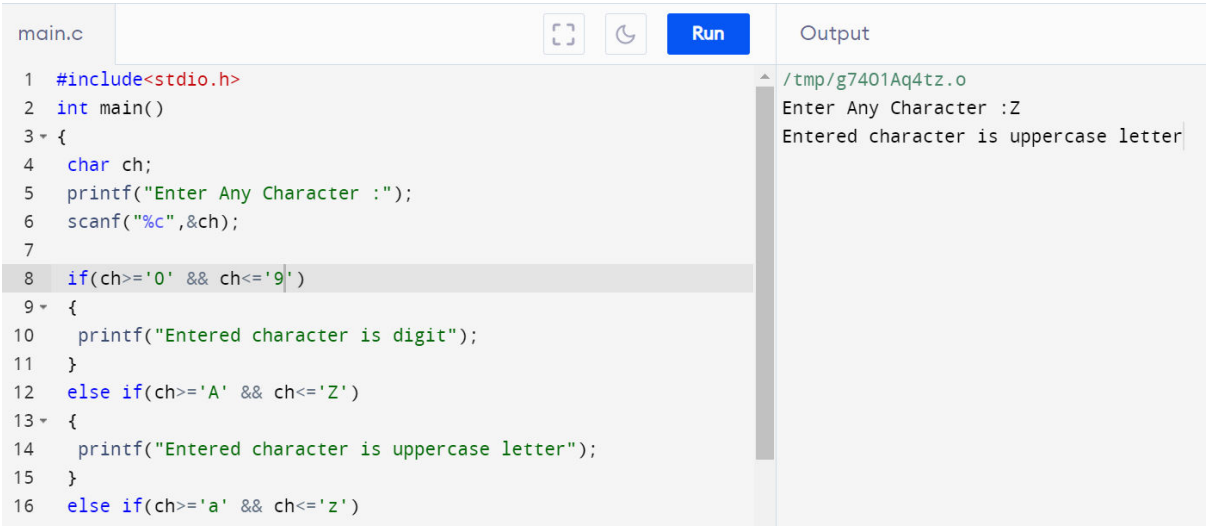
Lab 4 TakeHome Assignment

1.Write a program to find if an input character is a digit ,a lower case character, an upper case character or a special character.

```
#include<stdio.h>

int main()

{

 char ch;

 printf("Enter Any Character :");

 scanf("%c",&ch);


 if(ch>='0' && ch<='9')

 {

  printf("Entered character is digit");

 }

 else if(ch>='A' && ch<='Z')

 {

  printf("Entered character is uppercase letter");

 }

 else if(ch>='a' && ch<='z')

 {

  printf("Entered character is small letter");

 }

 else

 {
```

Samuela Abigail
71762108039

```
 printf("Entered character is special character");
 }
 return 0;
}
```



Inference- if negative integers are typed as the character, then output will come as special character instead of digit since – (minus) is a special character. It works fine for positive integers though.

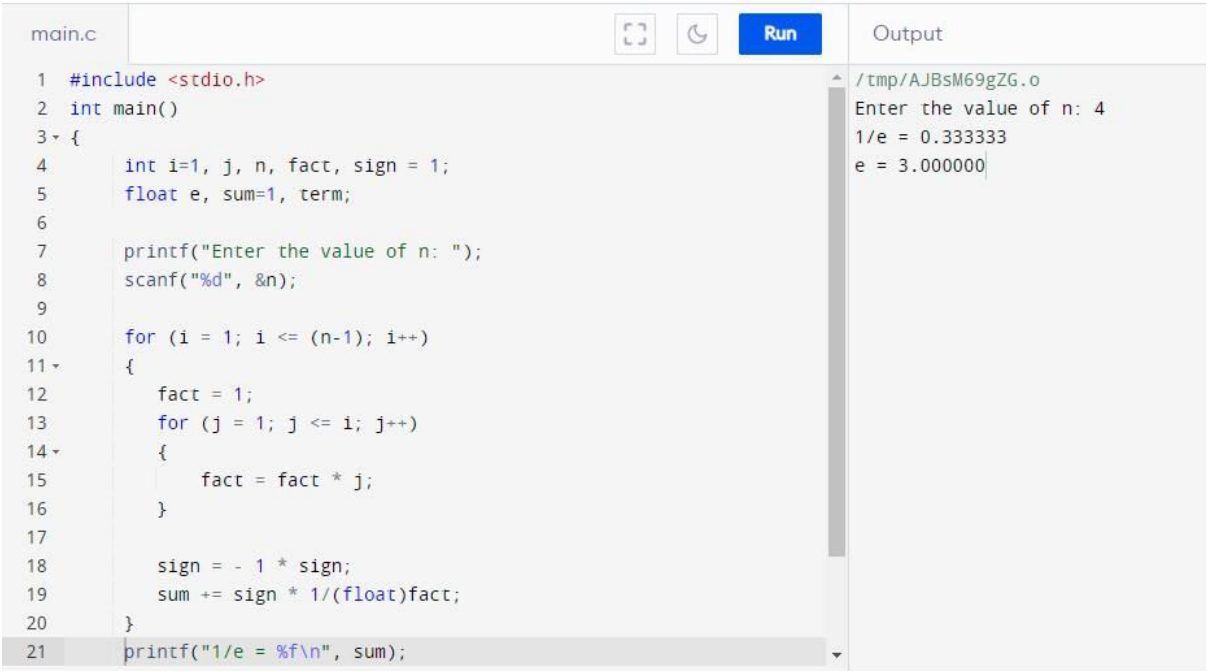
2.Write a program which reads an integer n, and finds the value of constant e using the following series truncated to n terms: 1/e = 1 – 1/1! + 1/2! –1/3! + ...

```
#include <stdio.h>
int main()
{
    int i=1, j, n, fact, sign = 1;
    float e, sum=1, term;

    printf("Enter the value of n: ");
```

```
    scanf("%d", &n);


    for (i = 1; i <= (n-1); i++)
    {
      fact = 1;
      for (j = 1; j <= i; j++)
      {
         fact = fact * j;
      }


      sign = - 1 * sign;
      sum += sign * 1/(float)fact;
    }
    printf("1/e = %f\n", sum);
    printf("e = %f", 1/sum);


    return 0;
}
```

Inference- Value of e is 2.71828 approximately. The program coded above gives an e value which is closer to 2.71828 for all possible values of n.

Samuela Abigail
71762108039



3.given two character arrays s1[25] and s2[25], check whether s2 is rotated string of s1 or not.

Ex: S1= tea; S2= eat; ----- Yes

S1 = Apple S2 = pleap; ---- Yes

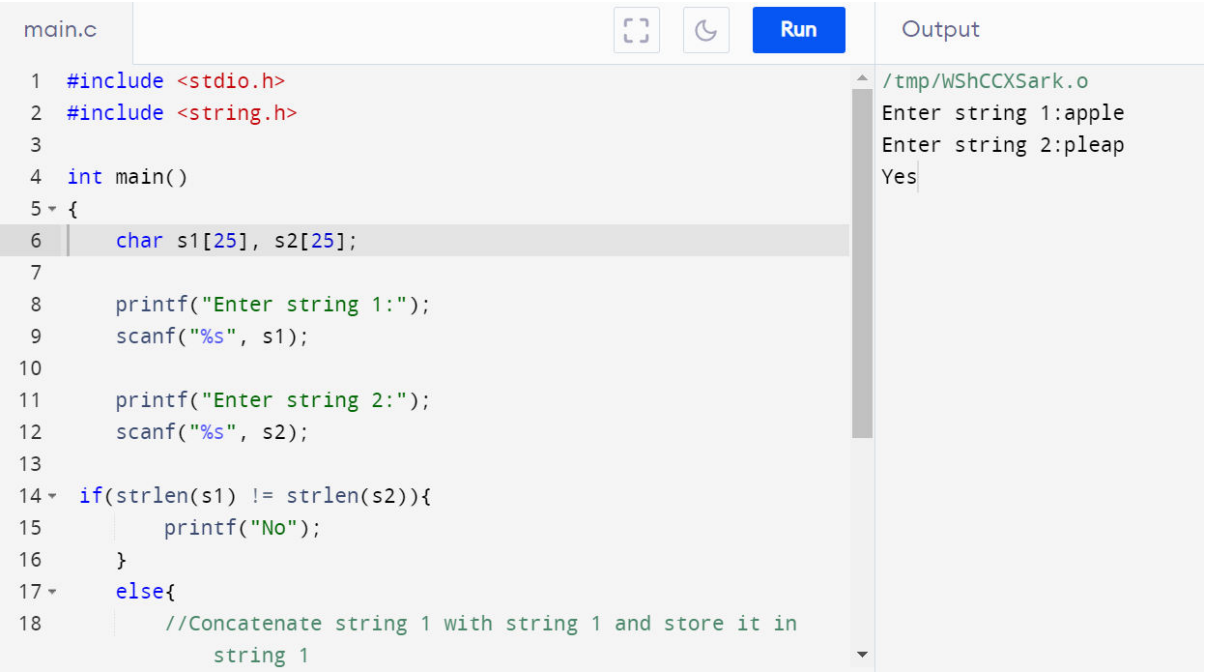
```
#include <stdio.h>
#include <string.h>


int main()
{
   char s1[25], s2[25];


   printf("Enter string 1:");
   scanf("%s", s1);
```

Samuela Abigail
71762108039

```
    printf("Enter string 2:");
    scanf("%s", s2);

 if(strlen(s1) != strlen(s2)){
        printf("No");
    }
    else{
        //Concatenate string 1 with string 1 and store it in string 1
        strcat(s1, s1);

        //Check whether string 2 is present in string 1
        if(strstr(s1, s2) != NULL)
            printf("Yes");
        else
            printf("No");
    }

    return 0;
}
```

Inference- string header file is used to code for this program.

Samuela Abigail
71762108039



4.Evaluate a given expression which is represented as a character array. Ex: 4+120-8 Output: 116

```
#include <stdio.h>

#include <string.h>

#include <stdlib.h>


int main()

{

int i, k=0, length, expsum=0;

char exptext[20], tempexp[20]="", op='+';


//Take input of the expression.

printf("\n");

printf("Enter a Experssion :");
```

```
gets(exptext);

length = strlen(exptext);//length of the expression string

for (i=0; i<=length-1; i++)
{
printf("character of expression string at %d is %c\n", i, exptext[i]);

if ((exptext[i]== '+') || (exptext[i]=='-'))
{
// Based on the operand expression adds or subtracts to the total sum
if (op == '+') expsum += atoi(tempexp);
if (op == '-') expsum -= atoi(tempexp);

k=0; // Reset temp array index controller
memset(tempexp, 0, 20); // Reset temp character array using memset function
op = exptext[i];// Save the operand for next operation
 }
 else
{
tempexp[k] = exptext[i];
 k=k+1;
```

```
}
}
// Calculation of the last part of the experssion
if (op == '+') expsum += atoi(tempexp);
if (op == '-') expsum -= atoi(tempexp);

printf("Result of the Expresion %s is %d\n", exptext, expsum);
return 0;
}
```

Inference- it works for + and – operands only. It won't work if the expression has more than 3 terms. The numerical terms of the expression are stored in an array, and the expression as a whole acts as a string of characters.

main.c

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int main()
{
int i, k=0, length, expsum=0;
char exptext[20], tempexp[20]="", op='+';

//Take input of the expression.
printf("\n");
printf("Enter a Experssion :");
gets(exptext);

length = strlen(exptext);//length of the expression string

for (i=0; i<=length-1; i++)
{
printf("character of expression string at %d is %c\n", i,
```

Run

Output

```
/tmp/qtyJLKVGXT.o
Enter a Experssion :5-9+56
character of expression string at 0 is 5
character of expression string at 1 is -
character of expression string at 2 is 9
character of expression string at 3 is +
character of expression string at 4 is 5
character of expression string at 5 is 6
Result of the Expresion 5-9+56 is 52
```

Samuela Abigail
71762108039