|  |  |
| --- | --- |
| **Title:** |  |
| **Problem Statement** | Write as per experiment list. |

**Programmer Name:**

**Batch:**

**Note: copy code and then program output (No screen shots) for all cases here**

**Input -**

#include <stdio.h>

int substring(char[], char[]);

int palindrome(char[]);

int compare(char[], char[]);

void copy(char[]);

void reverse(char[]);

int main()

{

printf("IMPORTANT :- String input should NOT contain any spaces. String length < 1000\n");

printf("OPTION MENU - \n");

printf("1. Substring (Checks if one string is a substring of the other.)\n");

printf("2. Palindrome (Checks if the given string is a palindrome. CASE SENSITIVE.)\n");

printf("3. Compare (Checks the equality of two strings)\n");

printf("4. Copy (Copies entered string into another string)\n");

printf("5. Reverse (Reverses the entered string)\n");

int input = 0;

printf("Select the operation that you want to perform :- ");

scanf("%d", &input);

switch (input)

{

case 1:

{

//Substring

char inparr1[1000];

char inparr2[1000];

printf("Enter primary string :- ");

scanf("%s", inparr1);

printf("Enter secondary string :- ");

scanf("%s", inparr2);

int answer = substring(inparr1, inparr2);

if(answer == 1) printf("'%s' is a substring of '%s'\n", inparr2, inparr1);

else printf("'%s' is not a substring of '%s'\n", inparr2, inparr1);

break;

}

case 2:

{

//Palindrome

char inparr1[1000];

printf("Enter string :- ");

scanf("%s", inparr1);

int answer = palindrome(inparr1);

if(answer == 1) printf("'%s' is a palindrome\n", inparr1);

else printf("'%s' is not a palindrome\n", inparr1);

break;

}

case 3:

{

//Compare

char inparr1[1000];

char inparr2[1000];

printf("Enter string 1 :- ");

scanf("%s", inparr1);

printf("Enter string 2 :- ");

scanf("%s", inparr2);

int answer = compare(inparr1, inparr2);

if(answer == 1) printf("'%s' is equal to '%s'\n", inparr2, inparr1);

else printf("'%s' is not equal to '%s'\n", inparr2, inparr1);

break;

}

case 4:

{

//Copy

char inparr[1000];

printf("Enter a string :- ");

scanf("%s", inparr);

copy(inparr);

break;

}

case 5:

{

//Reverse

char inparr[1000];

printf("Enter a string :- ");

scanf("%s", inparr);

reverse(inparr);

break;

}

default:

{

printf("Enter a valid option.\n");

break;

}

}

}

int substring(char arr1[], char arr2[])

{

int ret1 = 0;

int length1 = 0;

int length2 = 0;

for(length1 = 0; arr1[length1] != '\0'; ++length1);

for(length2 = 0; arr2[length2] != '\0'; ++length2);

for(int i = 0; i < length1; i++)

{

int ctr = 0;

for(int j = 0; j < length2; j++)

{

if(arr2[j] == arr1[i])

{

i++;

ctr++;

}

else

{

break;

}

}

if(ctr == length2)

{

ret1 = 1;

break;

}

}

return ret1;

}

int palindrome(char arr1[])

{

int length1 = 0;

int ret = 0;

for(length1 = 0; arr1[length1] != '\0'; ++length1);

int mid = length1/2;

int ctr = 0;

for(int i = 0; i < mid; i++)

{

if(arr1[i] != arr1[length1-i-1])

{

ctr++;

break;

}

}

if (ctr == 0)

{

ret = 1;

}

return ret;

}

int compare(char arr1[], char arr2[])

{

int ret = 0;

int length1 = 0;

int length2 = 0;

for(length1 = 0; arr1[length1] != '\0'; ++length1);

for(length2 = 0; arr2[length2] != '\0'; ++length2);

if(length1 == length2)

{

int ctr = 0;

for(int i = 0; i < length1; i++)

{

if(arr1[i] != arr2[i])

{

ctr++;

break;

}

}

if(ctr == 0)

{

ret = 1;

}

}

return ret;

}

void copy(char arr1[])

{

int length1 = 0;

for(length1 = 0; arr1[length1] != '\0'; ++length1);

char copyarr[length1];

for(int i = 0; i < length1; i++)

{

copyarr[i] = arr1[i];

}

printf("The original string is :- \n");

for(int i = 0; i < length1; i++)

{

printf("%c", arr1[i]);

}

printf("\n");

printf("The copied string is :- \n");

for(int i = 0; i < length1; i++)

{

printf("%c", copyarr[i]);

}

printf("\n");

}

void reverse(char arr1[])

{

int length1 = 0;

for(length1 = 0; arr1[length1] != '\0'; ++length1);

char revarr[length1];

for(int i = 0;i < length1; i++)

{

revarr[length1 - i - 1] = arr1[i];

}

printf("The original array is :- \n");

for(int i = 0; i < length1; i++)

{

printf("%c", arr1[i]);

}

printf("\nThe reversed array is :- \n");

for(int i = 0; i < length1; i++)

{

printf("%c", revarr[i]);

}

printf("\n");

}

**Output -**

**//Case 1 :- (Substring)**

IMPORTANT :- String input should NOT contain any spaces. String length < 1000

OPTION MENU -

1. Substring (Checks if one string is a substring of the other.)

2. Palindrome (Checks if the given string is a palindrome. CASE SENSITIVE.)

3. Compare (Checks the equality of two strings)

4. Copy (Copies entered string into another string)

5. Reverse (Reverses the entered string)

Select the operation that you want to perform :- 1

Enter primary string :- DataStructures

Enter secondary string :- Struct

'Struct' is a substring of 'DataStructures'

Do you wish to continue? [Y/n] Y

IMPORTANT :- String input should NOT contain any spaces. String length < 1000

OPTION MENU -

1. Substring (Checks if one string is a substring of the other.)

2. Palindrome (Checks if the given string is a palindrome. CASE SENSITIVE.)

3. Compare (Checks the equality of two strings)

4. Copy (Copies entered string into another string)

5. Reverse (Reverses the entered string)

Select the operation that you want to perform :- 1

Enter primary string :- Hello

Enter secondary string :- World

'World' is not a substring of 'Hello'

Do you wish to continue? [Y/n] Y

**//Case 2:- (Palindrome)**

IMPORTANT :- String input should NOT contain any spaces. String length < 1000

OPTION MENU -

1. Substring (Checks if one string is a substring of the other.)

2. Palindrome (Checks if the given string is a palindrome. CASE SENSITIVE.)

3. Compare (Checks the equality of two strings)

4. Copy (Copies entered string into another string)

5. Reverse (Reverses the entered string)

Select the operation that you want to perform :- 2

Enter string :- madam

'madam' is a palindrome

Do you wish to continue? [Y/n] y

IMPORTANT :- String input should NOT contain any spaces. String length < 1000

OPTION MENU -

1. Substring (Checks if one string is a substring of the other.)

2. Palindrome (Checks if the given string is a palindrome. CASE SENSITIVE.)

3. Compare (Checks the equality of two strings)

4. Copy (Copies entered string into another string)

5. Reverse (Reverses the entered string)

Select the operation that you want to perform :- 2

Enter string :- DataStructures

'DataStructures' is not a palindrome

Do you wish to continue? [Y/n] y

**//Case 3:- (Compare)**

IMPORTANT :- String input should NOT contain any spaces. String length < 1000

OPTION MENU -

1. Substring (Checks if one string is a substring of the other.)

2. Palindrome (Checks if the given string is a palindrome. CASE SENSITIVE.)

3. Compare (Checks the equality of two strings)

4. Copy (Copies entered string into another string)

5. Reverse (Reverses the entered string)

Select the operation that you want to perform :- 3

Enter string 1 :- Ma'am

Enter string 2 :- madam

'madam' is not equal to 'Ma'am'

Do you wish to continue? [Y/n] Y

IMPORTANT :- String input should NOT contain any spaces. String length < 1000

OPTION MENU -

1. Substring (Checks if one string is a substring of the other.)

2. Palindrome (Checks if the given string is a palindrome. CASE SENSITIVE.)

3. Compare (Checks the equality of two strings)

4. Copy (Copies entered string into another string)

5. Reverse (Reverses the entered string)

Select the operation that you want to perform :- 3

Enter string 1 :- Computers

Enter string 2 :- Computers

'Computers' is equal to 'Computers'

Do you wish to continue? [Y/n] y

**//Case 4 :- (Copy)**

IMPORTANT :- String input should NOT contain any spaces. String length < 1000

OPTION MENU -

1. Substring (Checks if one string is a substring of the other.)

2. Palindrome (Checks if the given string is a palindrome. CASE SENSITIVE.)

3. Compare (Checks the equality of two strings)

4. Copy (Copies entered string into another string)

5. Reverse (Reverses the entered string)

Select the operation that you want to perform :- 4

Enter a string :- Electronics

The original string is :-

Electronics

The copied string is :-

Electronics

Do you wish to continue? [Y/n] Y

**//Case 5 :- (Reverse)**

IMPORTANT :- String input should NOT contain any spaces. String length < 1000

OPTION MENU -

1. Substring (Checks if one string is a substring of the other.)

2. Palindrome (Checks if the given string is a palindrome. CASE SENSITIVE.)

3. Compare (Checks the equality of two strings)

4. Copy (Copies entered string into another string)

5. Reverse (Reverses the entered string)

Select the operation that you want to perform :- 5

Enter a string :- Telecommunication

The original array is :-

Telecommunication

The reversed array is :-

noitacinummoceleT

Do you wish to continue? [Y/n] n