Day 5: Strings(7-8-2025)

1. Write a program to find the length of a string without using strlen().

IPO:

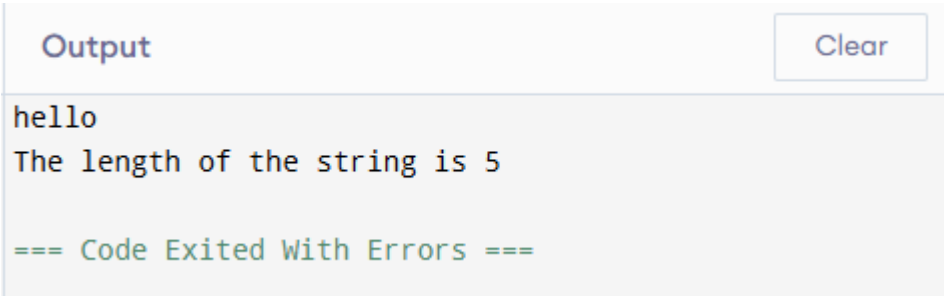
Input: string s

Process: using while loop incrementing the value of i for every letter

Output: length of the string

Code:

```
#include<stdio.h>
void main()
{
char s[10] = "hello", i=0;
while(s[i] != '\0')
i++;
printf("The length of the string is %d", i);
}
```



2. Write a program to copy one string to another.

IPO:

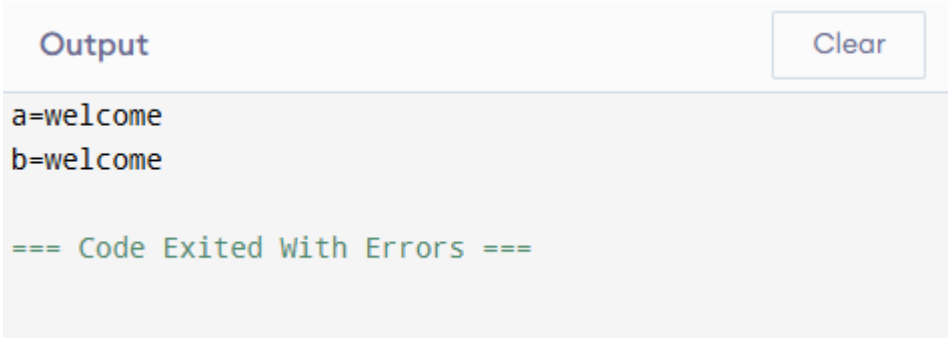
Input: string a

Process: using for loop copying element form a to b

Output: copying the string

Code:

```
#include<stdio.h>
void main()
{
char a[10] = "welcome", b[10], i;
printf("a=%s",&a);
for( i = 0 ; i<10;i++)
b[i] = a[i];
printf("\nb=%s",b);
}
```



3. Write a program to concatenate two strings.

IPO:

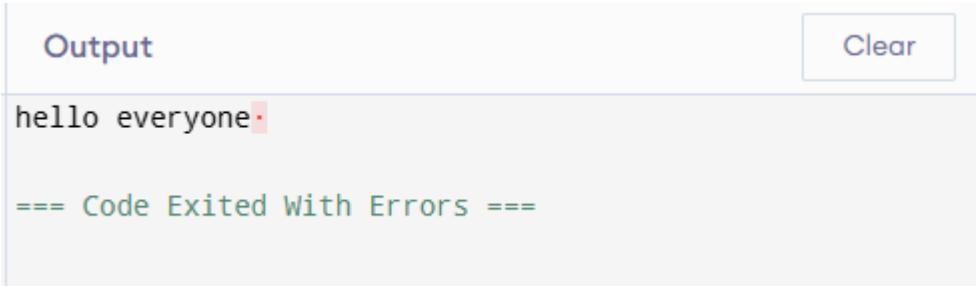
Input: two strings x and y

Process: using for loop add each element of x and y to z

Output: concatenate two strings

Code:

```
#include<stdio.h>
void main()
{
char x[6] = "hello",y[10] = " everyone", z[15],i,j=0;
for(i=0;i<15;i++)
{
if(i<5)
z[i] = x[i];
else
{
z[i] = y[j];
j++;
}
}
for(i=0;i<16;i++)
printf("%c", z[i]);
}
```



4. Write a program to compare two strings.

IPO:

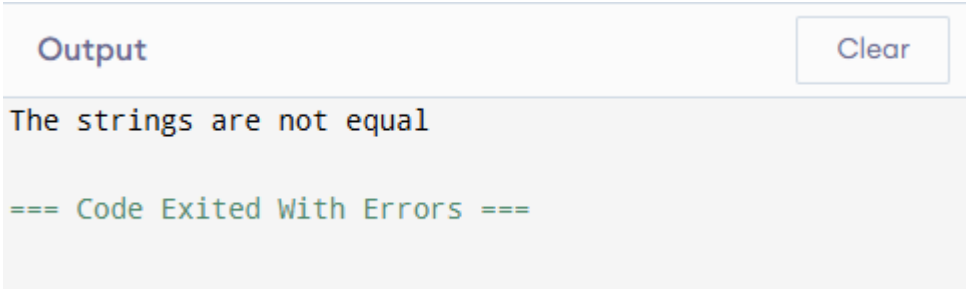
Input: two strings str1 and str 2

Process: using for loop compare each element of the two string

Output: compare the two strings

Code:

```
#include <stdio.h>
void main()
{
 char str1[5] = "hi";
 char str2[5] = "hello";
 int n=5,result = 0;
 for(int i=0;i<n;i++)
 {
 result = str1[i]-str2[i];
 }
 if (result == 0)
 printf("The strings are equal");
 else
 printf("The strings are not equal");
}
```



5. Write a program to count vowels and consonants in a string.

IPO:

Input: string say a

Process: using for loop compare each element of the string with the vowels and increment the

evecount

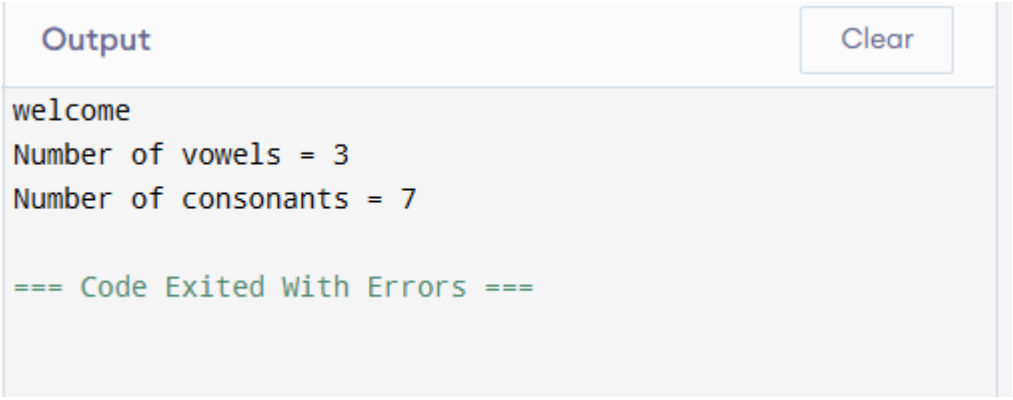
Else increment the concount

Output: count the number of vowels and consonants

Code:

```
#include<stdio.h>
void main()
{
char a[10] = "welcome" ,evecount = 0, concount = 0,i;
printf("%s",&a);
for (i=0;i<10 ;i++)
{
char c = a[i];
if(c == 'a' || c == 'e' || c == 'o' || c=='i' || c == 'u')
evecount++;
else
concount++;
}
printf("\nNumber of vowels = %d\nNumber of consonants = %d", evecount,concount);
```

}



6. Write a program to convert lowercase to uppercase and vice versa

IPO:

Input: string say c

Process: using for loop convert the lowercase letter to uppercase using ASCII value and

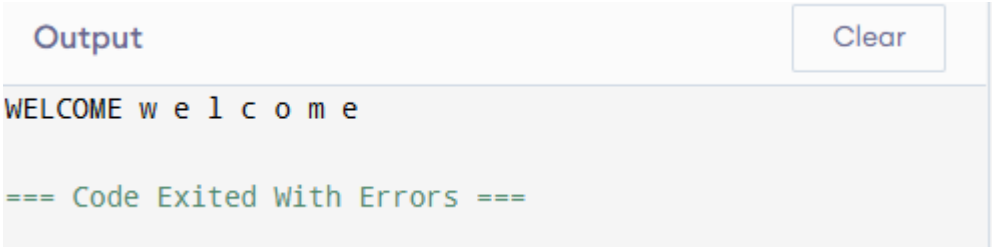
uppercase to lower case

Output: convert lowercase to uppercase

Code:

```
#include<stdio.h>
void main()
{
char c[10] = "WELCOME", i;
printf("%s",&c);
for(i=0;i<10;i++)
{
if(c[i] >='a' && c[i] <='z')
```

```
c[i]=c[i]-32;
else
c[i]=c[i]+32;
}
for (i=0;i<10;i++)
printf(" %c",c[i]);
}
```



7. Write a program to check if a string is palindrome.

IPO

Input: string say a

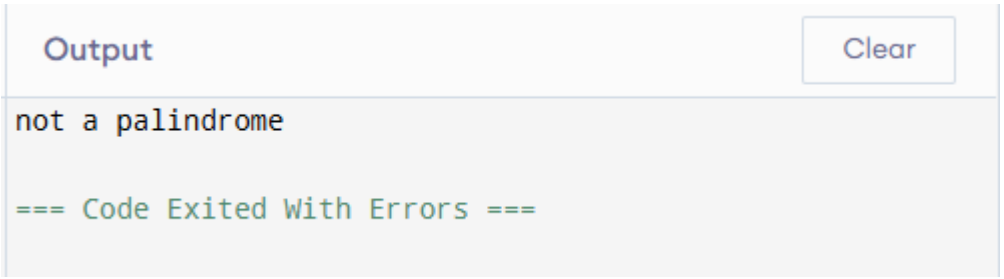
Process: using for loop reverse the string a and store it b and check whether the two are equal

Output: Check whether the string is palindrome

Code:

```
#include<stdio.h>
void main()
{
char a[5] = "hello", b[5], i, f=0;
for(i=0;i<5;i++)
b[i] = a[4 - i];
```

```
for( i = 0; i < 5 ;i++)
{
if(b[i] !=a[i])
{
f=1;
break;
}
}
if(f==0)
printf("palindrome");
else
printf("not a palindrome");
}
```



8. Write a program to reverse a string.

IPO

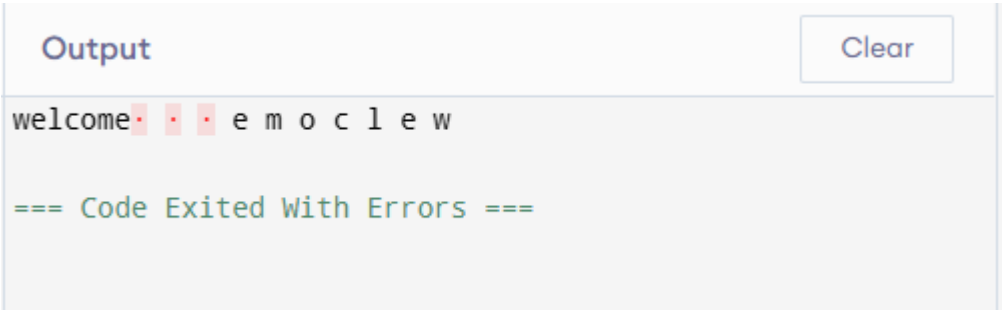
Input: string say x

Process: using for loop reverse the string a and store it in y

Output: reverse the string a

Code:

```
#include<stdio.h>
void main()
{
char x[10] ="welcome", y[10], i;
printf("%s",x);
for(i=0;i<10;i++)
y[i]=x[9-i];
for(i=0;i<10; i++)
printf("%c ",y[i]);
}
```



9. Write a program to count words in a string.

IPO

Input: string say a.

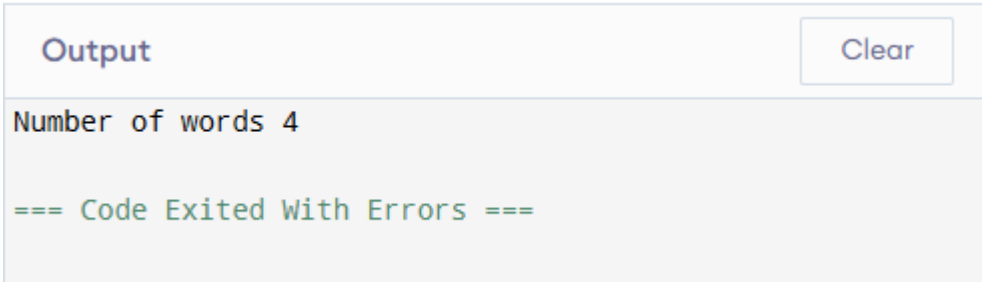
Process: using for loop count the number of spaces in the string by incrementing the value of

count and print the number of words , as count denotes the number of words.

Output: to count the number of words.

Code:

```
#include<stdio.h>
void main()
{
char a[25] = "welcome to C programming", count=0, i;
for( i = 0; i < 25 ;i++)
{
if(a[i] ==' ')
count++;
}
printf("Number of words %d", count+1);
}
```



10. Write a program to find the frequency of each character in a string.

IPO:

Input: string say x.

Process: using for loop calculating the frequency of each character in the string , simultaneously

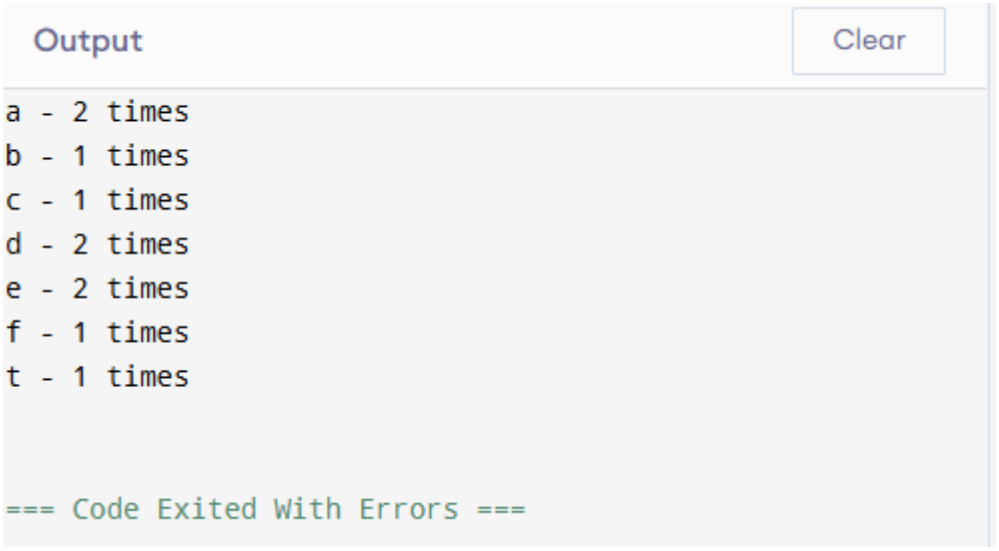
checking whether the character is already counted.

Output:frequency of each character in the string.

Code:

```
#include<stdio.h>
void main()
{
char x[10] = "abcdeadfet";
int i, j;
for(i = 0; i < 10 ; i++)
{
int c = 0;
for(j=0;j<i; j++)
{
if(x[i] == x[j])
{
c = 1;
break;
}
}
if(c == 1)
continue;
int count = 1;
for(j=i+1; j < 10 ;j++)
{
if(x[i] == x[j])
count++;
}
printf("%c - %d times\n", x[i], count);
```

```
}
}
```


Output
Clear
a - 2 times
b - 1 times
c - 1 times
d - 2 times
e - 2 times
f - 1 times
t - 1 times
=== Code Exited With Errors ===