VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PPSLAB

Name K. SREE INDIRA SIVANI Roll No. 1602-21-733-052 Page No. 119

PRELAB QUESTIONS-9:

1) List different functions available in string.h header file. Explain any 5 functions.

Ans: strlen → string length.
strcpy → string copy
strcmp → string comparison
strcat → string concatenate
strtok → locates first (or) next token in the string.
strchar → find character in string.
strstr → search for substring
strspn → spanning characters in the set.
strpbrk → returns pointer to first character.

* String length:
It returns the length of the specified string excluding the null character.
int strlen(char string);

* String copy:
Copies the contents of one string into another string. char strcpy(char tostr, char fromstr);

* String compare:
compares the string until it finds unequal characters (or) reaches the end of the string.
char → If 2 strings are equal : returns 0;
→ If first string is less than 2nd string; returns a value which is less than zero.
→ If first string is greater than 2nd string; returns a value greater than zero.

* String concatenation:

Joining the 2nd string to first string at the end or anywhere in between the 1st string.

```
char strcat (char s1, char s2, int size);
```

* character in string:

Used to find if the given character is present in the given string.

```
char strchr (char string, int ch);
```

2) Differentiate between scanf and gets:

Ans: Scanf is used to read the values from the keyboard; but this function terminates when it finds a space when only a format specifier is specified.

```
scanf("%c",&name);
```

Gets is used to read the values from the keyboard which also reads spaces and do not terminate until we press enter.

```
gets(name);
```

3) Define string. How to initialize a string?

Ans: String is a series of characters which are treated as 1 unit.

String initialization:

```
datatype stringname [size] = {'a','b','c'};
                        (or) = "abc";
```

4) What operations can be performed on strings?

Ans: ~~All arithmatic and relational operations~~
The string operations include concatenation, scanning, substringing, translation & verification.

5) Give the syntax and explain the functionality of dynamic memory management functions available in C.

Ans: SYNTAX:

```
int *p; → pointer variable
p = (datatype1*) malloc(n*sizeof(datatype2));
```

FUNCTIONALITY:

Dynamic memory allocation is the process of allocating the memory to the run-time variables during the execution of program.

PRELAB - PROGRAMS: 9

1) Programs to reverse the content of a given string without using predefined function:

* PROGRAM: <strops.c>

```
int stringlength(char a[])
 { int i; count = 0;
  for(i=0 ; a[i] != '\0'; i++)
    { count ++; }
```

```
return count;
}
# include <stdio.h>
# include "strops.c"
int main()
{ int len;
 char name[30];
 char rev[30];
 printf("Enter your name");
 scanf("%s1[^\n]",name);
   len = stringlength(name);
 for(j=0, i=len-1; i>=0; j++, i--)
  { rev[j]=name[i];
  } rev[j] = "\0";
 printf("The reverse is :");
 for(j=0; j<=0; j++)
 {
  printf("%s", rev);
 return 0;
 }.
```

→OUTPUT:

```
Enter your name:
SIVANI
The reverse is:
INAVIS
```

2) Simple program on pointer arithmetic operation:

* PROGRAM:

```
#include<stdio.h>
int main()
{int a[5]={10,20,30,40,50};
    int *p;
     p= &a[0];
     printf("%d",a[0]);
     printf("%d",*p);
       p++;
     printf("%d",*p);
          p+=3;
     printf("%d",*p);
        p=&a[1];
        p--;
      printf("%d",*p);
    return 0;
    }
```

→ OUTPUT:

```
10
10
20
40
10
```

3) Program to find the length of the given string by using user defined function:

```
#include<stdio.h> int slength(char a[]);
int main()
{  int length;
```

```
char name[30];
printf("Enter your name:");
scanf("%30[^\n]",name);
length = slength(name);
printf("Length of the string
                =%d", length);
return 0;
}
int slength(char a[]).
{  int i; count=0;
  for(i=0; a[i]!='\0'; i++).
   { count++; }
  return count;
  }
```

→OUTPUT:

```
Enter your name:
SIVANI
Length of the string
        = 6
```

4) Program to compare the contents of given 2 strings by using user defined functions:

```
# include <stdio.h>   ~~void~~ int compare (char, char);
int main ()
{ char string1[20];
  char string2[20];
   int i;
   gets(string1);
   gets(string2);
```

```
puts(string1);
puts(string2);
int l1, l2;
for(l1=0; string1[l1]!='\0'; l1++);
 for(l2=0; string2[l2]!='\0'; l2++);
  int min=l1<l2?l1:l2;
  int i;
 for(i=0; i<min; i++)
{ { if(s1[i]== if(string1[i]== string2[i])
                { x=0;
                  continue; }
              if(string1[i]> string2[i]) {
                { x=1;
                  break; }
               else { x=-1;
                      break; }.
  }
  if(x==0)
   { printf("The strings are same");
   }
if(x==+1)
   { printf("The first string is greater "); }
```

```
if(x==-1)
 { printf("The second string is greater");
  }
  return 0;
}
```

→ OUTPUT:

```
abcdef
pqrstu
abcdef
pqrstu
The second string is greater
```

5) Program to append the contents of the given string to another by using user defined function.

```
#include <stdio.h>
 int main()
{ char fname[10];
  char lname[10];
  char name[20];
  int i,j;
printf("Enter your first name:");
scanf("%s", fname);
printf("Enter your last name:");
 scanf("%s", lname);
```

```
for(i=0; fname[i]='\0';i++)
   { name[i]=fname[i];}
for(j=0; lname[j]='\0'; j++,i++)
     { name[i]=lname[j];}
       name[i]='\0';
      printf("%s",name);
return 0;
  }.
```

→ OUTPUT:

Enter your first name: INDIRA
Enter your last name: SIVANI
Name INDIRA SIVANI

→ OUTPUT:

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)

(Affiliated to Osmania University)

Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PPS LAB

Name K. SREE INDIRA SIVANI Roll No. 1602-21-733-052 Page No. 128.

→ AIM: Program to illustrate the use of strings

PROGRAM TO INSERT ONE STRING INTO ANOTHER:

→ PROBLEM STATEMENT: Write a program to copy the contents of a given string to another string at a specified location by using functions.

→ PROGRAM:

```
# include <stdio.h>
# include <stdlib.h>
int main()
{ char s1[m], s2[m];
   int n;
  printf("Enter the first string: ");
   gets(s1);
   printf("Enter the second string: ");
   gets(s2);
   printf("Enter the index where you want to insert:");
   scanf("%d",&n);
   int a,b;
   for(a=0; s1[a]!='\0'; a++);
   for(b=0; s2[b]!='\0'; b++);
   int i,j;
   for(i=a+b-1, j=a-1; j>=n; i--, j--)
```

```
{ s1[i] = s1[j]; }
int p,q;
for (p=0,q=n; p<b; p++,q++)
 { s1[q] = s2[p]; }
  printf("%s",s1);
  return 0;
}.
```

→ OUTPUT:

```
Enter the first string:
vasavi of engineering
Enter the second string:
college
Enter the index where you want to insert:
7
vasavi college of engineering.
```

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PPS LAB

Name K.SREE INDIRA SIVANI Roll No. 1602-21-733-052 Page No. 130

→ AIM: Program to illustrate the use of strings.
PRINT IF MOTHERTONGUE IS PALINDROME/NOT.

→ PROBLEM STATEMENT: Write a program to read your mother tongue and check whether it is a palindrome or not.

→ PROGRAM:

```
#include <stdio.h>
int slength(char a[]);
{ int i, count=0;
 for(i=0; a[i]!='\0' ; i++)
  { count++; }
  return count; }
void palindrome (char b[], char c[])
{ int i;
  for(i=0; i<slength(b) ; i++)
  { if (c[i] != b[i])
      break; }
  if (i==slength (b))
  printf ("PALINDROME");
  else
   printf("Not a Palindrome);
  }
```

```
int main()
{ int length, i, j;
   char original[30];
   char reverse[30];
   printf("Enter your mother tongue:");
   scanf("%.30[^\n]", original);
    length = slength(original);
   for(j=0;i=length-1; i>=0;j++,i--)
    { reverse[j]=original[i];
      }reverse[j]='\0';

   palindrome(original, reverse);
    return 0;
  }
```

→OUTPUT:

1) Enter your mother tongue: telugu.
Not a Palindrome

2) Enter your mother tongue: abcba.
PALINDROME

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PPSLAB

Name K.SREE INDIRASIVANI Roll No. 1602-21-733-052 Page No. 132.

AIM: Program to illustrate the use of strings.

PROGRAM TO READ AND SORT COLLEGE NAMES:

PROBLEM STATEMENT: Write a program to read the no. of colleges participating and also accept the names. Display the collegenames in alphabetical order.

PROGRAM:

```
#include<stdio.h>
#include<string.h>
int main()
{ int n,i,j;
  printf("Enter the number of colleges participating:");
  scanf("%d", &n);
  char cnames[50][50];
  char temp[50]; for(i=0;i<n;i++){scanf("%s",cnames[i]);}
  for(i=0; i<n; i++)
  { for (j=i+1; j<n ; j++)
    { if(strcmp(cnames[i], cnames[j])>0)
    { strcpy(temp, cnames[i]);
      strcpy(cnames[i], cnames[j]);
      strcpy(cnames[j], temp); } } }
printf("Colleges in alphabetical order}:\n");
 for(i=0; i<n; i++)
  { printf("%s\n", cnames[i]); }
  return 0; }
```

→ OUTPUT:

```
Enter the number of colleges participating : 5
cbit
vasavi
mvsr
anurag
bvrit
Colleges in alphabetical order:
anurag
bvrit
cbit
mvsr
vasavi
```

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PPS LAB

Name K.SREE INDIRA SIVANI Roll No. 1602-21-733-052 Page No. 133

→ AIM: Program to illustrate the use of strings.

DISPLAY THE TOP 3 CGPA:

→ PROBLEM STATEMENT: Write a program to dynamically allocate space for storing the CGPA of the students in the class & display the top 3CGPA.

→ PROGRAM:

```
#include <stdio.h>
#include <stdlib.h>
int main()
{ int n,i,j;
   float cgpa[20];
   float t;
   printf("Enter the no. of students : ");
   scanf("%d",&n);
   flood *p= (float *) calloc (n, sizeof(float));
   p=cgpa;
   printf("Enter the %d students CGPA:\n",n);
  for(i=0; i<n; i++)
   { scanf("%f", (p+i)); }
  for(i=0; i<n-1; i++)
  { for(j=0; j<n-1; j++)
    { if(cgpa[j]<cgpa[j+1])
```

```
        { t = cgpa[j];
          cgpa[j] = cgpa[j+1];
          cgpa[j+1] = t;
        } } }.
printf("The sorted CGPA is: \n");
for(i=0; i<n; i++)
  { printf("%f\t", *(p+i));
  } printf("\n");
  printf("%f, %f, %f", *(p+0), *(p+1), *(p+2));
  return 0;
}
```

→ OUTPUT:

Enter the no. of students : 5
Enter the 5 students CGPA :

9
10
7
8
5

The sorted CGPA is :
10.000000. 9.000000 8.000000 7.000000 5.000000

10.000000, 9.000000, 8.000000