**LAB Week 9**

**Name: Muhammad Talha Shafiq Choudhary**

**Reg#:SP21-BSE-008**

**Exercise 1.**

Input a character from user and print following about the input character.

* 1. Islower
  2. Isupper
  3. Isalpha
  4. Isdigit
  5. Ispunct
  6. Isspace

|  |
| --- |
| **Solution 1:**  **#include<stdio.h>**  **int main()**  **{**  **char ch;**  **printf("Enter a character:\n");**  **scanf("%c",&ch);**  **printf("The isolower value is: %d \n",islower(ch));**  **printf("====================================\n");**  **printf("The isupper value is: %d \n",isupper(ch));**  **printf("====================================\n");**  **printf("The isalpha value is: %d \n",isalpha(ch));**  **printf("====================================\n");**  **printf("The isdigit value is: %d \n",isdigit(ch));**  **printf("====================================\n");**  **printf("The ispunct value is: %d \n",ispunct(ch));**  **printf("====================================\n");**  **printf("The isspace value is: %d \n",isspace(ch));**  **return 0;**  **}** |

**Exercise 2.**

1. Write down a function called CountUpper, for counting upper case characters in some string e.g. "123 W. 42nd St.,NY,NY 10020-1095".
2. Write a program to convert all uppercase to lowercase and all lowercase to uppercase characters.

|  |
| --- |
| **Solution 2:**  **#include<stdio.h>**  **#include<string.h>**  **void countupper(char str[]);**  **void uppercase(char str[]);**  **void lowercase(char str[]);**  **int main()**  **{**  **char str[60]="123 W. 42nd St.,NY,NY 10020-1095";**  **countupper(str);**  **uppercase(str);**  **lowercase(str);**  **}**  **void countupper(char str[])**  **{**  **int count=0,i;**  **for(i=0;str[i]!='\0';i++)**  **{**  **if(str[i]>='A'&&str[i]<='Z')**  **{**  **++count;**  **}**  **}**  **printf("The number of uppercase characters are %d: \n",count);**  **printf("=============================================\n\n");**  **return 0;**  **}**  **void uppercase(char str[])**  **{**  **printf("The upper case string : %s\n",strupr(str));**  **printf("=============================================\n\n");**  **}**  **void lowercase(char str[])**  **{**  **printf("The lower case string : %s\n",strlwr(str));**  **printf("=============================================\n\n");**  **}** |

**Exercise 3.**

Convert the following strings into appropriate numeric variables. Hint: you have to call ato\* functions.

1. “0”
2. “3.14159”
3. “21 PGECHS”
4. “PGECHS 21”
5. “PGECHS21”
6. “3..14159”
7. “.987”
8. "124z3yu87"
9. "-3.4" – convert to integer
10. “e24.5”

|  |
| --- |
| **Solution no 3:**  **#include<stdio.h>**  **#include<stdlib.h>**  **int main()**  **{**  **char a[]="0";**  **char b[]="3.14159";**  **char c[]="21 PGECHS";**  **char d[]="PGECHS 21";**  **char e[]="PGECHS21";**  **char f[]="3..14159";**  **char g[]=".987";**  **char h[]="124z3yu87";**  **char i[]="-3.4";**  **char j[]="e24.5";**  **printf("Numeric value is %d\n",atoi(a));**  **printf("Numeric value is %d\n",atoi(b));**  **printf("Numeric value is %d\n",atoi(c));**  **printf("Numeric value is %d\n",atoi(d));**  **printf("Numeric value is %d\n",atoi(e));**  **printf("Numeric value is %d\n",atoi(f));**  **printf("Numeric value is %d\n",atoi(g));**  **printf("Numeric value is %d\n",atoi(h));**  **printf("Numeric value is %d\n",atoi(i));**  **printf("Numeric value is %d\n",atoi(j));**  **return 0;**  **}** |

**Exercise 4.**

Compare strings "ABCDE" and "ABCE" using strcmp.

Compare strings "ABCDE" and "" using strcmp.

Compare strings "Pakistan" and "Iran" using strcmp.

Compare strings "Pak" and "Iran" using strcmp.

Concatenate strings “Pak ” and “China” using strcat.

Concatenate strings “Pak ” and “China” and str1 = “ Economic Corridor” using strcat.

Concatenate strings “Pak ” and “China” using strcat and copy in str3 using strcpy.

|  |
| --- |
| **Solution no 4:**  **#include<stdio.h>**  **int main()**  **{**  **char s1[50]="ABCDE";**  **char s2[50]="ABCE";**  **int result;**  **result=strcmp(s1,s2);**  **if(result==0)**  **{**  **printf("\n\tCompare strings ABCDE and ABCE:Equal\n");**  **}**  **else**  **{**  **printf("\n\tCompare strings ABCDE and ABCE: Unequal\n");**  **printf("\t======================================");**  **}**  **char s3[50]="ABCDE";**  **char s4[50]=" ";**  **result=strcmp(s3,s4);**  **if(result==0)**  **{**  **printf("\n\tCompare strings ABCDE and " ":Equal");**  **}**  **else**  **{**  **printf("\n\tCompare strings ABCDE and " ": Unequal\n");**  **printf("\t======================================");**  **}**  **char s5[50]="Pakistan";**  **char s6[50]="Iran";**  **result=strcmp(s5,s6);**  **if(result==0)**  **{**  **printf("\n\tCompare strings Pakistan and Iran:Equal");**  **}**  **else**  **{**  **printf("\n\tCompare strings Pakistan and Iran: Unequal\n");**  **printf("\t======================================");**  **}**  **char s7[50]="Pak";**  **char s8[50]="Iran";**  **result=strcmp(s7,s8);**  **if(result==0)**  **{**  **printf("\n\tCompare strings Pak and Iran:Equal");**  **}**  **else**  **{**  **printf("\n\tCompare strings Pak and Iran: Unequal\n");**  **printf("\t======================================");**  **}**  **char str2[50]="Pak";**  **char str4[50]="China";**  **strcat(str2,str4);**  **printf("\n\tString after concatenate:%s\n",str2);**  **printf("\t======================================");**    **char str5[50]="Pak";**  **char str6[50]="China";**  **char str1[50]="Economic Corridor";**  **strcat(str5,str6);**  **strcat(str5,str1);**  **printf("\n\tString after concetinate:%s\n",str5);**  **printf("\t======================================");**    **char str7[50]="Pak";**  **char str8[50]="China";**  **char str3[50];**  **strcat(str7,str8);**  **strcpy(str3,str7);**  **printf("\n\tString after concetenate:%s\n",str3);**  **("\t======================================\n\n\n");**  **return 0;**  **}** |

**Exercise 5.**

Allocate memory for a string of 15 characters and assign “new string” to it. Print the string. Now, try following options:

1. Change the string to “another string” using assignment operator.
2. Change the string to “another string” using strcpy.

Note down whether the address changes in both cases or not.

|  |
| --- |
| **Solution no 5:**  **#include <stdio.h>**  **#include <string.h>**  **int main()**  **{**  **char str1[15]="new string";**  **char str2[15]="another string";**  **int i;**  **for (i = 0; i<=15; ++i)**  **{**  **str1[i] = str2[i];**  **}**  **str2[i] = '\0';**  **printf("\n\tString changed with operator is: %s", str2);**  **printf("\n\t========================================");**  **strcpy(str1,str2);**  **printf("\n\tString changed with strcpy: %s\n", str1);**  **return 0;**  **}** |