

Name: Ali hassan

Reg.no: Sp20-bse-013

Section: A

Submitted to: Prof Abdul Karim shahid

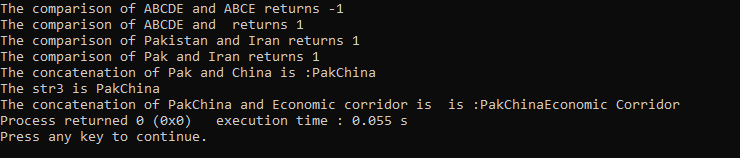
Exercise 1:

Description :This program is about comparing and concatenating different strings

Program:

|  |
| --- |
| #include<stdio.h>  #include<string.h>  **int** **main**(){  compare("ABCDE","ABCE");  compare("ABCDE","");  compare("Pakistan","Iran");  compare("Pak","Iran");  **char** y[**100**]="Pak";  **char** str3[**100**];  strcat(y,"China");  printf("The concatenation of Pak and China is :%s**\n**",y);  strcpy(str3,y);  printf("The str3 is %s**\n**",str3);  strcat(y,"Economic Corridor");  printf("The concatenation of PakChina and Economic corridor is is :%s",y);  }  **void** **compare**(**char** x[],**char** y[]){  printf("The comparison of %s and %s returns %d**\n**",x,y,strcmp(x,y));  } |

Output:



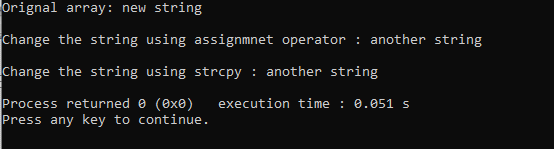
Exercise 2:

Description: This code is about assignment of strings using assignment operator and strcopy() method

Program:

|  |
| --- |
| #include<stdio.h>  #include<string.h>  **int** **main**(){  **char** s1[**15**]="new string";  **char** s2[**15**]="another string";  **char** \*p1=s1;  p1=s2;  printf("Orignal array: ");  puts(s1);  printf("**\n**Change the string using assignmnet operator : %s **\n**",p1);  strcpy(s1,s2);  printf("**\n**Change the string using strcpy : %s **\n**",s1);  **return** **0**;  } |

Output:



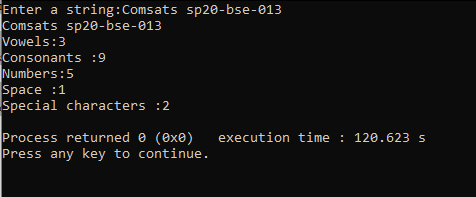
Exercise 3:

Description: This program counts vowels consonants numbers and other characters in an input string

Program:

|  |
| --- |
| #include<stdio.h>  #include<string.h>  **int** **main**(){  **int** numbers,vowels,consonants,space,other\_characters;  vowels=numbers=consonants=space=other\_characters=**0**;  **char** string[**100**];  printf("Enter a string:");  gets(string);  puts(string);  **for**(**int** i=**0**;string[i]!='\0';i++){  **if**(string[i]=='a'|| string[i]=='e'|| string[i]=='i'|| string[i]=='o'|| string[i]=='u'  || string[i]=='A'|| string[i]=='E'|| string[i]=='I'|| string[i]=='O'|| string[i]=='U')  {  vowels++;  }  **else** **if**((string[i]>='a'&& string[i]<='z') || (string[i]>='A'&& string[i]<='Z'))  {  consonants++;  }  **else** **if**(string[i]>='0' && string[i]<='9'){  numbers++;  }  **else** **if**(string[i]==' '){  space++;  }  **else**{  other\_characters++;  }  }  printf("Vowels:%d**\n**",vowels);  printf("Consonants :%d**\n**",consonants);  printf("Numbers:%d**\n**",numbers);  printf("Space :%d**\n**",space);  printf("Special characters :%d**\n**",other\_characters);  } |

Output:



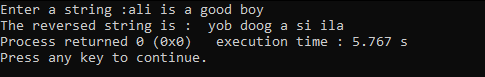
Exercise 4:

Description: This program returns a reversed string using recursion

Program:

|  |
| --- |
| #include<stdio.h>  **void** **reverse**(**char** \*p);  **int** **main**()  {  **char** str[**100**];  printf("Enter a string :");  gets(str);  printf("The reversed string is :**\n**");  reverse(str);  **return** **0**;  }  **void** **reverse**(**char** \*p){  **if**(p[**0**]=='\0')  **return**;  **else** reverse(&p[**1**]);  printf("%c",p[**0**]);  } |

Output:



Exercise 5:

Code:

|  |
| --- |
| #include<stdio.h>  #include<stdlib.h>  #include<string.h>  //execution of main  **int** **main**()  {  //declaration of string and initialization  **char** a[]="0";  **char** b[]="3.14159";  **char** c[]="21 PGECHS";  **char** \*cptr;  **char** d[]="PGECHS 21";  **char** dptr[**5**];  **char** e[]="PGECHS21";  **char** eptr[**5**];  **char** f[]=".987";  **char** g[]="124z3yu87";  **char** gptr[**7**];  **char** h[]="-34";  **char** i[]="24.5";  **int** j,jj;  //declaration to store the conversion in sebsequent data type  **int** x,p;  **long** **int** l;  **unsigned** **long** **int** o,m;  **double** k,n,q,y;  //conversion functions of string  x = atoi(a);  l = strtol(c,&cptr,**0**);  k = atof(b);  strncpy(dptr,&d[**7**],**3**);  m=atoi(dptr);  n = atof(f);  //o = strtoul(e,&eptr,0);  strncpy(eptr,&e[**6**],**3**);  o=atoi(eptr);  p = atol(h);  q = atof(i);  **for**(j=**0**,jj=**0**;g[j]!='\0';j++)  **if**(isdigit(g[j]))  gptr[jj++] = g[j];  gptr[jj] = '\0';  y = atof(gptr);  //output of the conversion  printf("String %s to int : %d**\n**",a,x);  printf("String %s to double : %f**\n**",b,k);  printf("String %s to long int : %u**\n**",c,l);  printf("String %s to unsigned long int : %u**\n**",d,m);  printf("String %s to unsigned long int : %u**\n**",e,o);  printf("String %s to double : %f**\n**",f,n);  printf("String %s to  long int : %d**\n**",h,p);  printf("String %s to double : %f**\n**",i,q);  printf("String %s to unsigned long int : %.0f**\n**",g,y);  **return** **0**;  } |

Output:

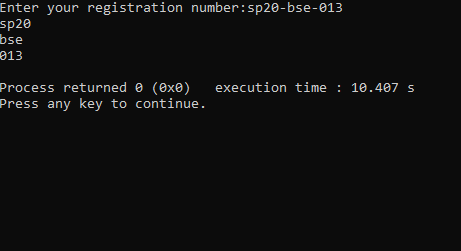
Exercise 6:

Description: This program separates batch roll. no in registration number using tokenization

Program:

|  |
| --- |
| #include<stdio.h>  #include<stdlib.h>  **int** **main**(){  **char** reg[**50**];  printf("Enter your registration number:");  gets(reg);  **char** \*token=strtok(reg,"-");  **while**(token!=NULL)  {  printf("%s**\n**",token);  token=strtok(NULL,"-");  }  } |

Output:

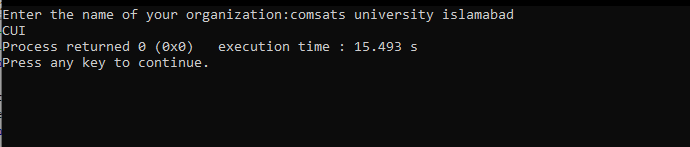


Exercise 7: This program finds the abbreviation of given name using string tokenization

Program:

|  |
| --- |
| #include<stdio.h>  #include<stdlib.h>  **int** **main**(){  **char** name[**50**];  printf("Enter the name of your organization:");  gets(name);  **char** \*token=strtok(name," ");  **while**(token!=NULL)  {  printf("%c",toupper(\*token));  token=strtok(NULL," ");  }  } |

Output:

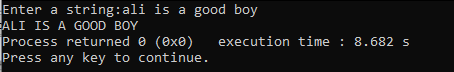


Exercise 8:

Description: This program return the given string all capetilized.

|  |
| --- |
| #include<stdio.h>  #include<string.h>  **int** **main**(){  **char** str[**100**];  printf("Enter a string:");  gets(str);  **for** (**int** i=**0**;i<strlen(str);i++){  printf("%c",toupper(str[i]));  }  } |

Output:

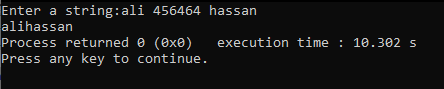


Exercise 9:

This program returns only alphabets in the given string.

|  |
| --- |
| #include<stdio.h>  #include<string.h>  **int** **main**(){  **char** str[**100**];  printf("Enter a string:");  gets(str);  **for**(**int** i=**0**;i<strlen(str);i++){  **if**(isalpha(str[i])){  printf("%c",str[i]);  } |

Output:

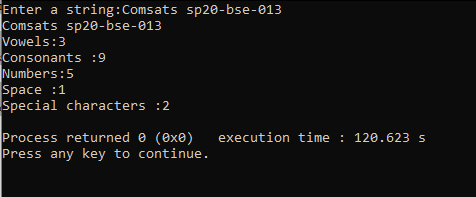


Exercise 10:

Part 1::

|  |
| --- |
| #include<stdio.h>  #include<string.h>  **int** **main**(){  **int** numbers,vowels,consonants,space,other\_characters;  vowels=numbers=consonants=space=other\_characters=**0**;  **char** string[**100**];  printf("Enter a string:");  gets(string);  puts(string);  **for**(**int** i=**0**;string[i]!='\0';i++){  **if**(string[i]=='a'|| string[i]=='e'|| string[i]=='i'|| string[i]=='o'|| string[i]=='u'  || string[i]=='A'|| string[i]=='E'|| string[i]=='I'|| string[i]=='O'|| string[i]=='U')  {  vowels++;  }  **else** **if**((string[i]>='a'&& string[i]<='z') || (string[i]>='A'&& string[i]<='Z'))  {  consonants++;  }  **else** **if**(string[i]>='0' && string[i]<='9'){  numbers++;  }  **else** **if**(string[i]==' '){  space++;  }  **else**{  other\_characters++;  }  }  printf("Vowels:%d**\n**",vowels);  printf("Consonants :%d**\n**",consonants);  printf("Numbers:%d**\n**",numbers);  printf("Space :%d**\n**",space);  printf("Special characters :%d**\n**",other\_characters);  } |

Output:



Part 2:: This program finds if a string contains only numbers, only alphabets, only alphanumeric characters,or mixed characters.

Code:

|  |
| --- |
| #include<stdio.h>  #include<string.h>  #include<stdbool.h>  **int** **main**(){  **bool** alphabet,number,alphanum,mixed;  printf("Enter a string:");  **char** str[**50**];  gets(str);  **for**(**int** i=**0**;i<strlen(str);i++){  **if**(str[i]==' '){  **continue**;  }  **else** **if**(!isalnum(str[i])){  mixed=true;  }  **else** **if**(isalpha(str[i])){  alphabet=true;  }  **else** **if**(isdigit(str[i])){  number=true;  }  }  **if**(mixed){  printf("Mixed");  }  **else** **if**(alphabet && number){  printf("Only Alphanumeric");  }  **else** **if**(alphabet){  printf("Only Alphabets");  }  **else** **if**(number){  printf("Numbers");  }  } |

Output:

