|  |  |  |
| --- | --- | --- |
| **Name:**  **P K Navin Shrinivas** | **SRN:PES2UG20CS237** | **Section:G(B-2)** |
| **Date:1/07/21** | **Week Number:9** |

|  |  |
| --- | --- |
| **1** | Write a C program to merge contents of two files into a third file.  Hint: Create three files- 2 files(file1.txt and file2.txt) with contents and third file(file3.txt) to merge contents of other two files(file1.txt and file2.txt).  Input:  gedit file1.txt  Hi,Good morning!  Have a nice day  gedit file2.txt  Welcome to C programming- file handling concepts  gedit file3.txt  //empty file  Enter the 1st file name : file1.txt  Enter the 2nd file name : file2.txt  Enter the new file name to merge the two files:file3.txt  Output:  The two files merged into file3.txt file successfully..!!  //Third file-Merged contents of two files(file1.txt and file2.txt)  gedit file3.txt  Hi,Good morning!  Have a nice day  Welcome to C programming- file handling concepts |
|  | **Program:**  **1file1.txt:**  $cat 1file1.txt  Hi,Good morning!  Have a nice day %  **1file2.txt:**  $cat 1file2.txt  Welcome to C programming- file handling concpets%  **1file3.txt [before merge]:**  $cat 1file3.txt  **1prog.c:**  #include<stdio.h>  int main() {  char file1[20], file2[20];  printf("Enter name of first file :");  scanf("%[^\n]%\*c", file1);  printf("Enter name of second file :");  scanf("%[^\n]%\*c", file2);  FILE \* f1 = fopen(file1, "r");  FILE \* f2 = fopen(file2, "r");  FILE \* f3 = fopen("1file3.txt", "w+");  char ch = fgetc(f1);  while (ch != EOF) {  if (ch != '\0')  fputc(ch, f3);  ch = fgetc(f1);  }  fputs("\n", f3);  ch = fgetc(f2);  while (ch != EOF) {  fputc(ch, f3);  ch = fgetc(f2);  }  printf("Two file merged into file3.txt successfully!");  fclose(f1);  fclose(f2);  fclose(f3);  }  **1file3.txt [After merge]:**  $cat 1file3.txt  Hi,Good morning!  Have a nice day  Welcome to C programming- file handling concpets% |
|  | **Output Screenshot:**  **1** |
| 2 | Write a C program to write multiple lines in a text file.  Input:  enter the filename  file.txt  Enter the number of lines to be written : 2  The lines are  hi hello  how are you  Output:  The content of the file file.txt is :  hi hello  how are you |
|  | **Program:**  **2file1.txt [before running program]:**  $cat 2file.txt  cat: 2file1.txt: No such file or directory  **2prog.c:**  #include<stdio.h>  #include<string.h>  int main() {  int n;  char file[50];  printf("Enter file name : ");  scanf("%[^\n]%\*c", file);  printf("Enter number of lines :");  scanf("%d", & n);  printf("Enter the lines : \n");  char data[n \* 1000];  fflush(stdin);  fflush(stdout);  for (int i = 0; i < n; i++) {  char new [] = "\n";  char temp[1000];  scanf(" %[^\n]%\*c", temp);  strcat(data, temp);  strcat(data, new);  }  FILE \* f = fopen(file, "w+");  fprintf(f, "%s", data);  printf("Contents of the file %s is : \n", file);  printf("%s", data);  }  **2file1.txt [After running program]:**  $cat 2file1.txt  hi hello  how are you |
|  | **Output Screenshot:**  **2** |
| 3 | Write a program to sort positive integers in the ascending order using insertion sort  Input:  Enter the number of elements u want to sort  5  Output:  Enter 5 elements  13  6  23  1  89  Before sorting  13  6  23  1  89  After sorting  1  6  13  23  89 |
|  | **Program:**  **3prog.c:**  #include<stdio.h>  void swap(int \* *a*, int *b*, int *c*) {  int temp = a[b];  a[b] = a[c];  a[c] = temp;  }  void insertsort(int \* *a*, int *n*) {  int p = 1, c;  while (p < n) {  int key = a[p];  for (int i = p; i > 0; i--) {  if (a[i] < key)  break;  else if (a[i - 1] > a[i])  swap(a, i, i - 1);  }  p++;  }  }  int main() {  int n;  printf("Enter number of element :");  scanf("%d", & n);  int num[n];  printf("Enter %d numbers :\n", n);  for (int i = 0; i < n; i++) {  scanf("%d", num + i);  }  printf("Number before sorting :\n");  for (int i = 0; i < n; i++) {  printf("%d ", \*(num + i));  }  printf("\n");  printf("Numbers after sorting :\n");  insertsort(num, n);  for (int i = 0; i < n; i++) {  printf("%d ", \*(num + i));  }  } |
|  | **Output Screenshot:**  **3** |
| 4 | Write a bubblesort program to sort students details based on students roll number/name in the ascending order using array of pointers, by taking input from csv file and using callback to call two functions i)sort based on roll number ii) sort based on name.  Input:  stud.csv file    Output:  99 xx  7 bb  22 cc  45 zz  8 aa  12 ff  4 gg  3 dd  27 jj  1 kk  32 ee  Enter your option  1.sort on roll  2.sort on name  0.exit  1  1 kk  3 dd  4 gg  7 bb  8 aa  12 ff  22 cc  27 jj  32 ee  45 zz  99 xx  Enter your option  1.sort on roll  2.sort on name  0.exit  2  8 aa  7 bb  22 cc  3 dd  32 ee  12 ff  4 gg  27 jj  1 kk  99 xx  45 zz |
|  | **Program:**  #include<stdio.h>  #include <stdlib.h>  #include<string.h>  void swap(char\* a[1][2],int *b*,int *c*)  {  char\* temp = a[b][0];  a[b][0]=a[c][0];  a[c][0]=temp;  temp=a[b][1];  a[b][1]=a[c][1];  a[c][1]=temp;  }  void bubbleroll(int n;char\* a[n][2] , int *n*)  {  int largest=0;  int larpos=0;  for(int i=0;i<n;i++)  {  for(int j=0;j<n-i-1;j++)  {  if(atoi(a[j][0])>atoi(a[j+1][0]))  {  swap(a,j,j+1);  }  }  }  }  void bubblename(int n;char\* a[n][2] , int *n*)  {  char\* largest=a[0][1];  int larpos=0;  for(int i=0;i<n;i++)  {  for(int j=0;j<n-i-1;j++)  {  if(strcmp(a[j][1],a[j+1][1])>0)  {  swap(a,j,j+1);  }  }  }  }  void csvsort(int n;void (\*func)(int n;char\* a[n][2] , int n),char\* a[n][2], int *n*)  {  func(a,n);  }  int main(){  FILE\* f=fopen("4file.csv","r");  char data[1000];  fscanf(f,"%[^NULL]%\*c",data);  char\* split=data;  char\* token1=strtok\_r(split,"\n",&split);  char\* raw[100][2];  int i=0;  while(token1!=NULL)  {  char\* token2=strtok(token1,",");  raw[i][0]=token2;  token2=strtok(NULL,",");  raw[i][1]=token2;  token1=strtok\_r(split,"\n",&split);  i++;  }  int flag=1;  while(flag)  {  int choice;  printf("0.Exit \n");  printf("1.Sort byroll No.\n");  printf("2.Sort by name \n");  printf("3.List contents of the file\n");  printf("Enter your option :");  scanf("%d",&choice);  if(choice==0)  return 0;  else if(choice == 1)  {  char\* copy[100][2];  for(int c=0;c<i;c++)  {  copy[c][0]=raw[c][0];  copy[c][1]=raw[c][1];  }  csvsort(bubbleroll,copy,i);  printf("List after sorting by roll number : \n");  for(int c=0;c<i;c++)  {  printf("%s %s\n",\*(\*(copy+c)+0),\*(\*(copy+c)+1));  }  }  else if(choice == 2)  {  char\* copy[100][2];  for(int c=0;c<i;c++)  {  copy[c][0]=raw[c][0];  copy[c][1]=raw[c][1];  }  csvsort(bubblename,copy,i);  printf("List after sorting by name : \n");  for(int c=0;c<i;c++)  {  printf("%s %s\n",\*(\*(copy+c)+0),\*(\*(copy+c)+1));  }  }  else if(choice == 3)  {  for(int c=0;c<i;c++)  {  printf("%s %s\n",\*(\*(raw+c)+0),\*(\*(raw+c)+1));  }  }  }  } |
|  | **Output Screenshot:**  **4_1**  **4_2**  **4_3**  **4_4** |