1. Klavyeden girilen bir veriyi yeni dosya oluşturup içine kaydedecek C program kodunu yazınız.

*#include <stdio.h>*

*#include <stdlib.h>*

*#include<stdbool.h>*

*#include <string.h>*

*// klavyeden gelen veriyi yeni dosyaya kaydet*

***int******main****()*

*{*

***char*** *inputString[****1000****];*

***FILE*** *\*filePointer;*

***char*** *fileName[****20****] = "test.txt";*

 *printf("****\n****Bir cumle yazınız :****\n****");*

 *filePointer = fopen(fileName, "w");*

***if****(filePointer ==NULL)*

 *{*

 *printf(" Dosya acilirken hata!");*

 *exit(****1****);*

 *}*

 *fgets(inputString,* ***sizeof*** *inputString, stdin);*

 *fprintf(filePointer, "%s", inputString);*

 *fclose(filePointer);*

 *printf("****\n*** Dosya basariyla olusturuldu. **\n**", fileName);

 **return** **0**;

2. Bir txt dosyayı okuyup içeriğini konsola yazacak C kodunu yazınız.

#include <stdio.h>

#include <stdlib.h>

#include<stdbool.h>

#include <string.h>

// dosyayi ac, oku konsola yaz

**int** **main**()

{

 **char** karakter;

 **FILE** \*filePointer;

 **char** fileName[**20**] = "test.txt";

 printf("**\n** test.txt dosyasi okunuyor...:**\n**");

 filePointer = fopen(fileName, "r");

 **if**(filePointer ==NULL)

 {

 printf(" Dosya acilirken hata!");

 exit(**1**);

 }

 karakter = fgetc(filePointer);

 **while** (karakter != EOF)

 {

 printf ("%c", karakter);

 karakter = fgetc(filePointer);

 }

 fclose(filePointer);

 printf("**\n** Dosya basariyla okundu. **\n**", fileName);

 **return** **0**;

}

3. Verilen bir txt dosyasında kaç satır olduğunu konsola yazan bir C program kodu yazınız.

#include <stdio.h>

#include <stdlib.h>

#include<stdbool.h>

#include <string.h>

// dosyadaki satir sayisini bul

**int** **main**()

{

 **char** karakter;

 **char** karakterOnceki = NULL;

 **FILE** \*filePointer;

 **char** fileName[**20**] = "test.txt";

 printf("**\n** test.txt dosyasindaki satir sayisi sayiliyor...:**\n**");

 filePointer = fopen(fileName, "r");

 **if**(filePointer ==NULL)

 {

 printf(" Dosya acilirken hata!");

 exit(**1**);

 }

 **int** satirSayisi = **0**;

 karakter = getc(filePointer);

 **while**(karakter != EOF) {

 **if** (karakter == '\n') {

 satirSayisi++;

 }

 karakterOnceki = karakter;

 karakter = getc(filePointer);

 }

 **if** (karakterOnceki != NULL) {

 satirSayisi++;

 }

 fclose(filePointer);

 printf("**\n** Dosyadaki satir sayisi: %d**\n**", satirSayisi);

 **return** **0**;

}

4. Verilen bir text dosyasının içindeki kelime ve karakter sayısını konsola yazan C kodunu yazınız.

#include <stdio.h>

#include <stdlib.h>

#include<stdbool.h>

#include <string.h>

// dosyadaki kelime ve karakter sayisini bul

**int** **main**()

{

 **char** karakter;

 **char** karakterOnceki = NULL;

 **FILE** \*filePointer;

 **char** fileName[**20**] = "test.txt";

 printf("**\n** test.txt dosyasindaki kelime ve karakter sayisi sayiliyor...:**\n**");

 filePointer = fopen(fileName, "r");

 **if**(filePointer ==NULL)

 {

 printf(" Dosya acilirken hata!");

 exit(**1**);

 }

 **int** kelimeSayisi = **0**;

 **int** karakterSayisi = **0**;

 karakter = getc(filePointer);

 **while**(karakter != EOF) {

 karakterSayisi++;

 **if** (karakter == ' ' || karakter == '\n') {

 kelimeSayisi++;

 }

 karakterOnceki = karakter;

 karakter = getc(filePointer);

 }

 **if** (karakterOnceki != NULL) {

 kelimeSayisi++;

 }

 fclose(filePointer);

 printf("**\n** Dosyadaki karakter sayisi: %d**\n**", karakterSayisi);

 printf("**\n** Dosyadaki kelime sayisi: %d**\n**", kelimeSayisi);

 **return** **0**;

}

5. Verilen bir metin dosyasında istenen satırı silip dosyayı kaydeden C kodunu yazınız.

#include <stdio.h>

#include <stdlib.h>

#include<stdbool.h>

#include <string.h>

// dosyada istenen satiri sil

**int** **main**()

{

 **FILE** \*filePointer;

 **FILE** \*filePointerGecici;

 **char** fileName[**20**] = "test.txt";

 **char** geciciDosya[] = "gecici.txt";

 **int** bufferMiktari = **500**;

 **char** buffer[bufferMiktari];

 **int** silinecekSatir = **0**;

 printf("**\n** test.txt dosyasindaki istediginiz satir silinecek...:**\n**");

 printf(" Lutfen silmek istediginiz satir nosunu giriniz : ");

 scanf("%d", &silinecekSatir);

 filePointer = fopen(fileName, "r");

 **if**(filePointer ==NULL)

 {

 printf(" Dosya acilirken hata!");

 exit(**1**);

 }

 filePointerGecici = fopen(geciciDosya, "w");

 **if**(filePointerGecici ==NULL)

 {

 printf("Gecici Dosya acilirken hata!");

 exit(**1**);

 }

 **int** index = **0**;

 **do**

 {

 strcpy(buffer, "**\0**");

 fgets(buffer, bufferMiktari, filePointer);

 **if** (!feof(filePointer))

 {

 index++;

 **if** (index != silinecekSatir)

 {

 fprintf(filePointerGecici, "%s", buffer);

 }

 }

 } **while** (!feof(filePointer));

 fclose(filePointer);

 fclose(filePointerGecici);

 remove(fileName);

 rename(geciciDosya, fileName);

 printf("**\n** Dosyadaki %d. satir silindi!**\n**", silinecekSatir);

 **return** **0**;

}

6. Verilen bir metin dosyasında istenen satırdaki metni klavyeden girilen yeni satır ile değiştirip dosyayı kaydeden C kodunu yazınız.

#include <stdio.h>

#include <stdlib.h>

#include<stdbool.h>

#include <string.h>

// dosyada istenen satiri degistir

**int** **main**()

{

 **FILE** \*filePointer;

 **FILE** \*filePointerGecici;

 **char** fileName[**20**] = "test.txt";

 **char** geciciDosya[] = "gecici.txt";

 **int** bufferMiktari = **500**;

 **char** buffer[bufferMiktari];

 **char** yeniSatir[**250**];

 **int** silinecekSatir = **0**;

 printf("**\n** test.txt dosyasindaki istediginiz satir degistirilecek...:**\n**");

 printf("**\n** Lutfen yeni satir icerigini giriniz : **\n**");

 fgets(yeniSatir, **sizeof** yeniSatir, stdin);

 printf(" Lutfen degistirmek istediginiz satir nosunu giriniz : ");

 scanf("%d", &silinecekSatir);

 filePointer = fopen(fileName, "r");

 **if**(filePointer ==NULL)

 {

 printf(" Dosya acilirken hata!");

 exit(**1**);

 }

 filePointerGecici = fopen(geciciDosya, "w");

 **if**(filePointerGecici ==NULL)

 {

 printf("Gecici Dosya acilirken hata!");

 exit(**1**);

 }

 **int** index = **0**;

 **do**

 {

 strcpy(buffer, "**\0**");

 fgets(buffer, bufferMiktari, filePointer);

 **if** (!feof(filePointer))

 {

 index++;

 **if** (index != silinecekSatir)

 {

 fprintf(filePointerGecici, "%s", buffer);

 } **else** {

 fprintf(filePointerGecici, "%s", yeniSatir);

 }

 }

 } **while** (!feof(filePointer));

 fclose(filePointer);

 fclose(filePointerGecici);

 remove(fileName);

 rename(geciciDosya, fileName);

 printf("**\n** Dosyadaki %d. satir degistirildi!**\n**", silinecekSatir);

 **return** **0**;

}

7. Klavyeden girilen satırları mevcut bulunan bir txt dosyasının sonuna ekleyen C program kodu yazınız. Klavyeden EXIT girilmesi ile programı sonlandırınız.

#include <stdio.h>

#include <stdlib.h>

#include<stdbool.h>

#include <string.h>

// dosyanin sonuna ekle

**int** **main**()

{

 **FILE** \*filePointer;

 **char** fileName[**20**] = "test.txt";

 **int** bufferMiktari = **500**;

 **char** buffer[bufferMiktari];

 printf(" Dosyaya eklemek istediklerinizi giriniz. Cikmak icin EXIT yaziniz: **\n**");

 filePointer = fopen(fileName, "a");

 **if**(filePointer == NULL)

 {

 printf(" Dosya acilirken hata!");

 exit(**1**);

 }

 **char** \* exitCase = "EXIT**\n**";

 **int** eklenenSatirSayisi = **0**;

 **do**

 {

 fgets(buffer, bufferMiktari, stdin);

 **if**(strcmp(exitCase, buffer) == **0**){

 **break**;

 }

 fprintf(filePointer, "%s", buffer);

 eklenenSatirSayisi++;

 } **while** (strcmp(exitCase, buffer) != **0**);

 fclose(filePointer);

 printf("**\n** Dosyaya %d adet satir eklendi!**\n**", eklenenSatirSayisi);

 **return** **0**;

}

8. Mevcut bulunan bir metin dosyasını klavyeden girilen yeni bir isimli dosyaya kopyalayınız.

#include <stdio.h>

#include <stdlib.h>

#include<stdbool.h>

#include <string.h>

// dosyayi baska isimli dosyaya kopyala

**int** **main**()

{

 **FILE** \*filePointer;

 **FILE** \*filePointerYeni;

 **char** fileName[**20**] = "test.txt";

 **char** yeniDosyaAdi[**50**];

 **int** bufferMiktari = **500**;

 **char** buffer[bufferMiktari];

 **char** yeniSatir[**250**];

 printf("**\n** Lutfen yeni dosya ismini giriniz : **\n**");

 fgets(yeniDosyaAdi, **sizeof** yeniDosyaAdi, stdin);

 **int** i = **0**;

 **for**(i = **0**; i < **50**; i++) {

 **if**(yeniDosyaAdi[i] == '\n') {

 yeniDosyaAdi[i] = '\0';

 }

 }

 filePointer = fopen(fileName, "r");

 **if**(filePointer ==NULL)

 {

 printf(" Dosya acilirken hata!");

 exit(**1**);

 }

 filePointerYeni = fopen(yeniDosyaAdi, "w");

 **if**(filePointerYeni == NULL)

 {

 printf("Yeni Dosya acilirken hata!");

 exit(**1**);

 }

 **int** index = **0**;

 **do**

 {

 strcpy(buffer, "**\0**");

 fgets(buffer, bufferMiktari, filePointer);

 **if** (!feof(filePointer))

 {

 index++;

 fprintf(filePointerYeni, "%s", buffer);

 }

 } **while** (!feof(filePointer));

 fclose(filePointer);

 fclose(filePointerYeni);

 printf("**\n** Dosya kopyalandi!**\n**");

 **return** **0**;

}

9. Klavyeden adı girilen mevcut bulunan bir txt dosyasını sistemden silen C kodu yazınız.

#include <stdio.h>

#include <stdlib.h>

#include<stdbool.h>

#include <string.h>

// adi verilen dosyayi sil. Dikkatli kullaniniz. Veri kayiplarinda sorumnluluk kabul edilmemektedir.

**int** **main**()

{

 **char** silinecekDosyaAdi[**50**];

 printf("**\n** Lutfen silinecek dosya ismini giriniz : **\n**");

 fgets(silinecekDosyaAdi, **sizeof** silinecekDosyaAdi, stdin);

 **int** i = **0**;

 **for**(i = **0**; i < **50**; i++) {

 **if**(silinecekDosyaAdi[i] == '\n') {

 silinecekDosyaAdi[i] = '\0';

 }

 }

 **int** result = remove(silinecekDosyaAdi);

 **if**(result == **0**) {

 printf("**\n** Dosya silindi !**\n**");

 } **else** {

 printf("**\n** Dosya Silinemedi!**\n**", result);

 }

 **return** **0**;

}