**Manipulisane fajlovima**

|  |  |
| --- | --- |
| MOD | OPIS |
| ios::**in** | Otvara fajl za čitanje |
| ios::**out** | Otvara fajl za pisanje i briše zatečeni sadržaj |
| ios::**app** | Novi sadržaj dodaje na kraj fajla |
| ios::**ate** | Otvara fajl i pomjera pokazivač na kraj fajla |
| ios::**trunc** | Ako fajl postoji, njegov sadržaj se odbacuje – briše |
| ios::**\_Nocreate** | Zahtijevani fajl mora postojati (neće biti automatski kreiran) |
| ios::**binary** | Otvara fajl u binarnom modu (podaci se ne prevode) |

|  |  |
| --- | --- |
| Funkcija | OPIS |
| **seekg()** | Na zadatu lokaciju pomjera pokazivač za čitanje sadržaja fajla, a koristi se sa ifstream objektom |
| **tellg()** | Vraća (očitava) poziciju pokazivača za čitanje |
| **seekp()** | Na zadatu lokaciju pomjera pokazivač za upisivanje sadržaja fajla, a koristi se sa ofstream objektom |
| **tellp()** | Vraća (očitava) poziciju pokazivača za upisivanje |

# Zadatak

Dovršite funkcije snimiUFajl i procitajIzFajl.

|  |
| --- |
| #include <iostream>  #include <fstream>  #include <stdlib.h>  using namespace std;  struct univerzitet  {  char naziv[30];  };  struct fakultet  {  char naziv[30];  univerzitet univerzitet;  };  struct student  {  int broj\_indeksa;  char ime\_prezime[30];  fakultet fakultet;  };  student unosStudenta()  {  student s;  cout << "Unesite ime i prezime studenta: " << endl;  cin.getline(s.ime\_prezime, 30);  cout << "Unesite broj indeksaa: " << endl;  cin >> s.broj\_indeksa;  cin.ignore();  cout << "Unesite naziv univerziteta: " << endl;  cin.getline(s.fakultet.univerzitet.naziv, 30);  cout << "Unesite naziv fakulteta: " << endl;  cin.getline(s.fakultet.naziv, 30);  return s;  }  void printStudenta(student& s)  {  cout << "\n============STUDENT============\n";  cout << "Ime studenta je: \t\t" << s.ime\_prezime << endl;  cout << "Broj indexa je: \t\t" << s.broj\_indeksa << endl;  cout << "Naziv univerziteta je: \t" << s.fakultet.univerzitet.naziv << endl;  cout << "Naziv fakulteta je: \t\t" << s.fakultet.naziv << endl;  }  void snimiUFajl(char imeFajla[], student student1)  {  }  void procitajIzFajla(char imeFajla[], student& s)  {    }  void main()  {  //kreiramo objekat kojim cemo upisati sadrzaj u fajl  student s1 = unosStudenta();    char imeFajla[255];  cout << "Unesite ime fajla: " << endl;  cin.getline(imeFajla, 255);  snimiUFajl(imeFajla, s1);  student s2;  procitajIzFajla(imeFajla, s2);  printStudenta(s2);      system("pause");  } |

*Rješenje*

|  |
| --- |
| void snimiUFajl(char imeFajla[255], student student1)  {  //otvaramo fajl za upis u binarnom formatu - ios::binary mod  //fout - skracenica od file output  ofstream fout(imeFajla, ios::out | ios::binary);  cout << "Objekat STUDENT1 upisujem u fajl: " << imeFajla << endl;  //objekat student1 koristeci funkciju write() upisujemo u fajl  fout.write((char\*)(&student1), sizeof(student1));  fout.close();  cout << "Sadrzaj fajla " << imeFajla << " upisujem u objekat STUDENT2" << endl;  }  void procitajIzFajla(char imeFajla[255], student& s)  {  //otvaramo fajl za ispis binarnog sadrzaja  //fin - skracenica od file input  ifstream fin(imeFajla, ios::in | ios::binary);  //sadrzaj fajla, koristeci funkciju read(), iscitavamo u objekat s  fin.read((char\*)(&s), sizeof(s));  fin.close();  } |

# Zadatak

Učitati sadržaj matrice iz tekstualnog fajla u dvodimenzionalni niz. Korisnik mora unijeti veličinu matrice.

# Zadatak

1. Implementirajte funkciju izmjeni. Funkcija treba kopirate sve karaktere iz fajla imeFajlaSource u fajl imeFajlaDest, tako što će sve karaktere koji predstavljaju mala slova engleske abecede pretvoriti u velika slova. Funkcija treba vratiti broj izmijenjenih karaktera.

|  |
| --- |
| int izmjeni(char\* imeFajlaSource, char\* imeFajlaDest); |

1. Implementirajte funkciju main koja će omogućiti korisniku da unese nazive fajlova i koja će pozvati funkciju izmjeni.

|  |  |
| --- | --- |
| Rješenje | |
| 1:  2:  3:  4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14:  15:  16:  17:  18:  19:  20:  21:  22:  23:  24:  25:  26:  27:  28:  29:  30:  31:  32:  33:  34:  35:  36:  37:  38:  39:  40:  41: | #include <iostream>  #include <fstream>  using namespace std;  int izmjeni(char\* imeFajlaSource, char\* imeFajlaDest)  {  ifstream f1(imeFajlaSource);  ofstream f2(imeFajlaDest);  char a;  int b = 0;  while(f1.get(a))  {  if (islower(a))  b++;  char v = toupper(a);  f2 << v;  }  f1.close();  f2.close();  return b;  }  void main()  {  char imeFajlaSource[30];  char imeFajlaDest[30];  cout << "Source file: ";  cin >> imeFajlaSource;  cout << "Destination file: ";  cin >> imeFajlaDest;  int i = izmjeni(imeFajlaSource, imeFajlaDest);  cout << "Broj izmjenjenih karaktera je : " << i << endl;  system("pause");  } |

# Zadatak

Implementirajte program sa funkcijom void dodajKomentar(char\* imeFajla, char\* komentar) koja će dodati na kraj fajla komentar kojeg korisnik unese.

|  |
| --- |
| Rješenje |
| #include <iostream>  #include <fstream>  #include <stdlib.h>  using namespace std;  void dodajKomentar(char\* imeFajla, char\* komentar)  {  ofstream fout(imeFajla, ios::app | ios::\_Nocreate);  fout << endl << komentar;  fout.close();  }  void main()  {  system("dir \*.txt \*.h \*.cpp");  char imeFajla[30];  cout << "File: ";  cin >> imeFajla;  char komentar[300];  cout << "Komentar: ";  cin >> komentar;  dodajKomentar(imeFajla, komentar);  system("pause");  } |

# Zadatak

Dodajte u prethodnom zadatku provjeru o uspješnosti otvaranja fajla.

# Zadatak

Dovršiti program

|  |
| --- |
| #include <iostream>  #include <fstream>  using namespace std;  #include <stdlib.h>  char \* crt = "\n==========================================================\n";  char \* poruka\_o\_gresci = "Greska prilikom otvaranja fajla!";  void UnosSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::UNOS SADRZAJA::" << crt;  const int max = 200;  char tekst[max];  //...  }  void PrikazSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::PRIKAZ SADRZAJA::" << crt;  //...  }  void DodavanjeSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::DODAVANJE SADRZAJA::" << crt;  const int max = 200;  char noviTekst[max];  //fajl mora postojati - ios::\_Nocreate  //...  }  void KopiranjeSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::KOPIRANJE SADRZAJA::" << crt;  const int max = 30;  char nazivNovogFajla[max];  //...  }  int VelicinaFajla(const char \* nazivFajla)  {  cout << crt << "\t\t::VELICINA FAJLA::" << crt;  //...    }  void PretragaSadrzajaKarakter(const char \* nazivFajla)  {  cout << crt << "\t\t::PRETRAGA::" << crt;  char trazeni;  cout << "Unesite znak koji trazite: ";  cin >> trazeni;  //...  }  void PretragaSadrzajaRijec(const char \* nazivFajla)  {  cout << crt << "\t\t::PRETRAGA::" << crt;  char trazeni[50];  cout << "Unesite rijec koju trazite: ";  cin.getline(trazeni, 50);  //...  }  void BrisanjeSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::BRISANJE SADRZAJA::" << crt;  //...  }  //da li je izbor trebao biti povratna vrijednost?  void prikaziMeni(int & izbor)  {  do  {  cout << crt << "\t\t::MANIPULISANJE FAJLOVIMA::" << crt;  cout << "1. Upisi sadrzaj u fajl. " << endl;  cout << "2. Prikazi sadrzaj fajla. " << endl;  cout << "3. Dodaj novi sadrzaj fajlu. " << endl;  cout << "4. Kopiraj sadrzaj fajla u drugi fajl. " << endl;  cout << "5. Pretrazuj sadrzaj fajla (karakter). " << endl;  cout << "6. Pretrazuj sadrzaj fajla (rijec). " << endl;  cout << "7. Prikazi velicinu fajla. " << endl;  cout << "8. Brisi sadrzaj fajla. " << endl;  cout << "9. Rad sa novim fajlom. " << endl;  cout << "10. Izadji iz programa. " << crt;  cout << "Unesite vas izbor: ";  cin >> izbor;  std::cin.ignore();  system("cls");  } while (izbor<1 || izbor>10);  }  void main()  {  int izbor = 1;  const int max = 30;  char nazivFajla[max];  do  {  cout << crt << "\t\t::MANIPULISANJE FAJLOVIMA::" << crt;  cout << "Unesite ime fajla i ekstenziju: ";  cin.getline(nazivFajla, max);  do  {  prikaziMeni(izbor);  switch (izbor)  {  case 1:  UnosSadrzaja(nazivFajla); break;  case 2:  PrikazSadrzaja(nazivFajla); break;  case 3:  DodavanjeSadrzaja(nazivFajla); break;  case 4:  KopiranjeSadrzaja(nazivFajla); break;  case 5:  PretragaSadrzajaKarakter(nazivFajla); break;  case 6:  PretragaSadrzajaRijec(nazivFajla); break;  case 7:  VelicinaFajla(nazivFajla); break;  case 8:  BrisanjeSadrzaja(nazivFajla); break;  }  } while (izbor != 9 && izbor != 10);  } while (izbor != 10); system("pause");  } |

**Rješenje – Preuzeto iz PR-II-V**

|  |
| --- |
| #include <iostream>  #include <fstream>  using namespace std;  #include <stdlib.h>  char \* crt = "\n==========================================================\n";  char \* poruka\_o\_gresci = "Greska prilikom otvaranja fajla!";  void UnosSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::UNOS SADRZAJA::" << crt;  const int max = 200;  char tekst[max];  ofstream upis(nazivFajla);  if (upis.fail())  cout << crt << poruka\_o\_gresci << crt;  else  {  cout << "Unesite tekst: ";  cin.getline(tekst, max);  upis << tekst;  cout << crt << "Sadrzaj upisan u fajl!" << crt;  }  upis.close();  }  void PrikazSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::PRIKAZ SADRZAJA::" << crt;  char znak;  //zahtjevani fajl mora postojati jer koristimo ios::\_Nocreate mod  ifstream ispis(nazivFajla, ios::in | ios::\_Nocreate);  if (ispis.fail())  cout << crt << poruka\_o\_gresci << crt;  else  {  cout << crt << "FAJL: " << nazivFajla << crt << "SADRZAJ:" << crt;  while (ispis.get(znak))  cout << znak;  cout << crt << "Sadrzaj fajla prikazan!" << crt;  }  ispis.close();  }  void DodavanjeSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::DODAVANJE SADRZAJA::" << crt;  const int max = 200;  char noviTekst[max];  //fajl mora postojati - ios::\_Nocreate  //novi sadrzaj ce biti dodat na kraj fajla - ios::app  ofstream upis(nazivFajla, ios::app | ios::\_Nocreate);  if (upis.fail())  cout << crt << poruka\_o\_gresci << crt;  else  {  cout << "Unesite tekst koji zelite dodati: ";  cin.getline(noviTekst, max);  upis << endl << noviTekst << endl;  cout << crt << "Novi sadrzaj uspjesno dodat." << endl;  }  upis.close();  }  void KopiranjeSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::KOPIRANJE SADRZAJA::" << crt;  const int max = 30;  char nazivNovogFajla[max];  char znak;  cout << "Unesite ime fajla u koji zelite kopirati sadrzaj: ";  cin.getline(nazivNovogFajla, max); //otvaramo fajl odakle kopiramo  ifstream ispis(nazivFajla, ios::in | ios::\_Nocreate);  //otvaramo fajl gdje kopiramo  ofstream upis(nazivNovogFajla,ios::app);  if (!ispis.fail())  {  //provjeravmo validnost izvornog fajla  if(!upis.fail())  {//provjeravmo validnost destinacijskog fajla  while(ispis.get(znak))//uzimamo naredni karakter  upis << znak;//i upisujemo ga u destinacijski fajl  cout << crt << "Sadrzaj fajla: " << nazivFajla;  cout << " uspjesno dodat fajlu: " << nazivNovogFajla << crt;  }  else  cout << crt << "Destinacijski::" << poruka\_o\_gresci << crt;  }  else  cout << crt << "Izvorni::" << poruka\_o\_gresci << crt;  ispis.close();  upis.close();  }  int VelicinaFajla(const char \* nazivFajla)  {  cout << crt << "\t\t::VELICINA FAJLA::" << crt;  ifstream ispis(nazivFajla);  if (!ispis.fail())  {  //pomijeramo pokazivač na kraj fajla  ispis.seekg(0,ios::end);  //očitavamo poziciju pokazivača, sto predstavlja broj bajta  int velicina = ispis.tellg();  cout<<"Velicina fajla je: "<<velicina<<" bajta."<<crt;  ispis.close();  return velicina;  }  else  cout << crt << poruka\_o\_gresci << crt;  return -1;  }  void PretragaSadrzajaKarakter(const char \* nazivFajla)  {  cout << crt << "\t\t::PRETRAGA::" << crt;  char znak, trazeni;  int ukupno = 0, pronadjeno = 0;  ifstream ispis(nazivFajla);  cout << "Unesite znak koji trazite: ";  cin >> trazeni;  cin.ignore(100, '\n');  if (!ispis.fail())  {  while (ispis.get(znak))  {  if (znak == trazeni)  pronadjeno++;  ukupno++;  }  cout << crt << "Fajl " << nazivFajla << " ima ukupno ";  cout << ukupno << " znakova.";  cout << crt << "Znak '" << trazeni << "' se nalazi na ";  cout << pronadjeno << " mjesta.";  }  else  cout << crt << poruka\_o\_gresci << crt; ispis.close();  }  void PretragaSadrzajaRijec(const char \* nazivFajla)  {  //1. sadrzaj fajla ucitati u niz alociran u dinamickoj memoriji  //2. korisniku omoguciti da unese rijec koju trazi  //3. izvrsiti pretragu ucitanog sadrzaja fajla (niza),  // te o rezulatima pretrage obavijestiti korisnika  cout << crt << "\t\t::PRETRAGA::" << crt;  char trazeni[50];  cout << "Unesite rijec koju trazite: ";  cin.getline(trazeni, 50);  ifstream f(nazivFajla);  if(!f.fail())  {  char\* strF = new char[VelicinaFajla(nazivFajla) + 1];  char c;  int b = 0;  while(f.get(c))  {  strF[b] = c;  b++;  }  strF[b] = '\0';  char\* r = strstr(strF, trazeni);  if (r != nullptr)  {  cout << "Pronađeno na: " << r << endl;  }  }  }  void BrisanjeSadrzaja(const char \* nazivFajla)  {  cout << crt << "\t\t::BRISANJE SADRZAJA::" << crt;  ofstream ispis(nazivFajla, ios::trunc);  //isto bismo postigli da smo umjesto ios::trunc napisali ios::out  if(!ispis.fail())  cout << crt << "Sadrzaj fajla " << nazivFajla << " obrisan!" << crt;  else  cout << crt << poruka\_o\_gresci << crt;  }  //da li je izbor trebao biti povratna vrijednost?  void prikaziMeni(int & izbor)  {  do  {  cout << crt << "\t\t::MANIPULISANJE FAJLOVIMA::" << crt;  cout << "1. Upisi sadrzaj u fajl. " << endl;  cout << "2. Prikazi sadrzaj fajla. " << endl;  cout << "3. Dodaj novi sadrzaj fajlu. " << endl;  cout << "4. Kopiraj sadrzaj fajla u drugi fajl. " << endl;  cout << "5. Pretrazuj sadrzaj fajla (karakter). " << endl;  cout << "6. Pretrazuj sadrzaj fajla (rijec). " << endl;  cout << "7. Prikazi velicinu fajla. " << endl;  cout << "8. Brisi sadrzaj fajla. " << endl;  cout << "9. Rad sa novim fajlom. " << endl;  cout << "10. Izadji iz programa. " << crt;  cout << "Unesite vas izbor: ";  cin >> izbor;  std::cin.ignore();  system("cls");  } while (izbor<1 || izbor>10);  }  void main()  {  int izbor = 1;  const int max = 30;  char nazivFajla[max];  do  {  cout << crt << "\t\t::MANIPULISANJE FAJLOVIMA::" << crt;  cout << "Unesite ime fajla i ekstenziju: ";  cin.getline(nazivFajla, max);  do  {  prikaziMeni(izbor);  switch (izbor)  {  case 1:  UnosSadrzaja(nazivFajla); break;  case 2:  PrikazSadrzaja(nazivFajla); break;  case 3:  DodavanjeSadrzaja(nazivFajla); break;  case 4:  KopiranjeSadrzaja(nazivFajla); break;  case 5:  PretragaSadrzajaKarakter(nazivFajla); break;  case 6:  PretragaSadrzajaRijec(nazivFajla); break;  case 7:  VelicinaFajla(nazivFajla); break;  case 8:  BrisanjeSadrzaja(nazivFajla); break;  }  } while (izbor != 9 && izbor != 10);  } while (izbor != 10); system("pause");  } |

Autor:

mr. Adil Joldić

[adil@edu.fit.ba](mailto:adil@edu.fit.ba)