

Temat: Zapis do pliku i odczyt z pliku –podstawowe funkcje.

#include <fstream> - dyrektywa ta dołącza do programu bibliotekę, która zawiera funkcję do obsługi plików.
ofstream zapis („wyjście”); -deklaracja identyfikatora strumienia wyjściowego o nazwie zapis.

Tworzenie nowego pliku:

```
#include <iostream>
#include <cstdlib>
#include <fstream>
using namespace std;
int main()
{ ofstream zapis("wyjście.txt"); zapis<<3<<" "<<-20; zapis.close();
  return 0;
}
```

zapis.close();-zapisze i zamknie plik powiązany z obiektem zapis

zapis.open(“nazwa_pliku”); -otwarcie istniejącego pliku do zapisu, to co było zapisane w pliku zostaje skasowane

Tworzenie dwóch nowych plików

```
#include <iostream>
#include <fstream>
using namespace std;
int main()
{ int a=4;
  float b=8.4;
  ofstream wyjście("plik1.txt");      wyjście<<a<<endl<<b;      wyjście.close();
  wyjście.open("plik1.txt");          wyjście.close();
  wyjście.open("plik2.txt");          wyjście<<endl<<endl<<126;  wyjście.close();
  cout<<"koniec i sprawdź folder...";
  return 0;
}
```

Podawanie nazwy plików z klawiatury

```
#include <iostream>
#include <fstream>
using namespace std;
int main()
{ char nazwa[50];
  cout<<"Podaj nazwę pliku:"<<endl;  cin>>nazwa;
  ofstream plik(nazwa); for(int i=1;i<=10;i++) plik<<i; plik.close();
  cout<<"koniec i sprawdź folder...";
  return 0;
}
```

UWAGA!!!

Możemy również stosować zapis z użyciem ścieżek bezwzględnych i względnych:

```
ofstream wynik('c:/wyniki/rezultat.txt');
```

```
ofstream wynik('../rezulata.txt');
```

foldery wpisane w takich ścieżkach muszą już istnieć

Obsługa błędu otwarcia pliku do zapisu:

```
#include <iostream>
#include <fstream>
using namespace std;
int main()
{ ofstream wyniki("wynik.txt");
  if(!wyniki) { cout<<"Pliku nie można otworzyć"; getch(); return 1;}
  wyniki.close();
  cout<<"koniec i sprawdź folder...";
  return 0;
}
```

ifstream odczyt („wejscie.txt”); -deklaracja identyfikatora strumienia wejściowego o nazwie odczyt.

Przykład 1 Załóżmy, że w pliku o nazwie prostokat.txt zapisane są wymiary prostokąta w dwóch wierszach. Napisz program, który odczyta dwie liczby z pliku prostokat.txt i obliczy pole prostokąta o bokach wczytanych z pliku. Napisz też obsługę błędów, gdyby nie powiodła się próba odczytu z pliku

```
#include <iostream>
#include <fstream>
using namespace std;
int main()
{ float a,b;
  ifstream wejscie("prostokat.txt");
  wejscie>>a>>b;
  wejscie.close();
  cout<<"a= "<<a<<endl;
  cout<<"b= "<<b<<endl;
  cout<<"Pole prostokata wynosi: "<<a*b;
return 0;
}
```

Przykład 2 Napisz program który obliczy sumę liczb typu float zapisanych w pliku tekstowym o nazwie liczby.txt. Każda liczba jest w osobnej linii, nie wiemy ile jest liczb.

liczby.eof() –służy do wykrycia końca pliku , przyjmuje wartość prawda gdy jest koniec pliku lub fałsz w przeciwnym razie.

```
#include <iostream>
#include <fstream>
using namespace std;

int main()
{ float a,suma=0;
  ifstream liczby("liczby.txt");
  while (! liczby.eof())
  {   liczby>>a;
      suma=suma+a;   }
  liczby.close();
  cout<<"Suma wynosi: "<<suma;
  return 0;}
```

Przykład 3 Napisz program, który zliczy ile jest znaków (liter lub cyfr) w pliku tekstowym o nazwie tekst.txt.

tekst.get(zn) –wczytuje pojedynczy znak do zmiennej zn z pliku
isalnum(zn) – przyjmuje wartość prawda gdy zn jest literą lub cyfrą
isdigit(zn) –przyjmuje wartość prawda gdy zn jest liczbą
isalpha(zn) – przyjmuje wartość prawda gdy zn jest literą

```
#include <iostream>
#include <fstream>
using namespace std;

int main()
{ int n=0;
  char zn;
  ifstream tekst("tekst.txt");
  while (! tekst.eof())
  {   tekst.get(zn);
      if (isalnum(zn)) n=n+1;   }
  tekst.close();
  cout<<"ilosc znakow: "<<n;
  return 0;
}
```

Przykład 4 Zapisz tylko litery z pliku tekst.txt do pliku literki.txt
tekst.put(zn) –zapisuje do pliku pojedynczy znak

```
#include <iostream>
#include <fstream>
using namespace std;

int main()
{ char zn;
  ifstream tekst("tekst.txt");
  ofstream litery("literki.txt");
  while (! tekst.eof())
  { tekst.get(zn);
    if (isalpha(zn) ) litery.put(zn);    }
  tekst.close();
  litery.close();
return 0;}
```

Podsumowanie:

#include <fstream> - dyrektywa ta dołącza do programu bibliotekę, która zawiera funkcję do obsługi plików.
ofstream zapis („wyjście”); -deklaracja identyfikatora strumienia wyjściowego o nazwie zapis.
zapis.close();-zapisze i zamknie plik powiązany z obiektem zapis
zapis.open(“nazwa_pliku”); -otwarcie istniejącego pliku do zapisu, to co było zapisane w pliku zostaje skasowane
ifstream odczyt („wejście.txt”); -deklaracja identyfikatora strumienia wejściowego o nazwie odczyt.
liczby.eof() –służy do wykrycia końca pliku , przyjmuje wartość prawda gdy jest koniec pliku lub fałsz w przeciwnym razie.
tekst.get(zn) –wczytuje pojedynczy znak do zmiennej zn z pliku
isalnum(zn) – przyjmuje wartość prawda gdy zn jest literą lub cyfrą
isdigit(zn) –przyjmuje wartość prawda gdy zn jest liczbą
isalpha(zn) – przyjmuje wartość prawda gdy zn jest literą
tekst.put(zn) –zapisuje do pliku pojedynczy znak

Zadanie domowe

Zad.1. Napisz program który obliczy iloczyn liczb typu float zapisanych w pliku tekstowym o nazwie liczby.txt. Każda liczba jest w osobnej linii, nie wiemy ile jest liczb.

Zad.2. Napisz program który obliczy ile jest liczb całkowitych zapisanych w pliku tekstowym o nazwie liczby.txt. Każda liczba jest w osobnej linii..

Zad.3. Napisz program, który przepisze zawartość pliku liczby1.txt do pliku liczby2.txt. Tak aby każda liczba była w osobnym wierszu.

Zad. 4(matura_pp_2013)

W pliku napisy.txt znajduje się 1000 napisów o długościach od 2 do 16 znaków, każdy napis w osobnym wierszu. W każdym napisie mogą wystąpić jedynie dwa znaki: „0” lub „1”.W wybranym przez siebie języku programowania **napisz program**, za pomocą którego uzyskasz odpowiedzi na poniższe polecenia. Odpowiedzi zapisz w pliku zadanie4.txt, a odpowiedź do każdego podpunktu poprzedź literą oznaczającą ten podpunkt.

a) Podaj, ile jest napisów o parzystej długości.

b) Podaj, ile jest napisów, które zawierają taką samą liczbę zer i jedynek.

c) Podaj, ile jest napisów składających się z samych zer, oraz podaj, ile jest napisów składających się z samych jedynek.

d) Dla każdej liczby $k = 2, 3, \dots, 16$ podaj liczbę napisów o długości k znajdujących się w pliku napisy.txt, tzn. podaj, ile jest napisów 2-znakowych, ile jest napisów 3-znakowych itd.

Do oceny oddajesz plik(i) o nazwie(ach), zawierający(e) komputerową realizację Twojego rozwiązania, oraz plik tekstowy zadanie4.txt, zawierający odpowiedzi do podpunktów a, b, c i d zadania (odpowiedź do każdego podpunktu poprzedź literą oznaczającą ten podpunkt).

Zad. 5(matura_pr_2013)

W pliku dane.txt znajduje się 5 000 liczb całkowitych z przedziału od 108 do 20000008, zapisanych w systemie ósemkowym, po jednej liczbie w wierszu. W wybranym przez siebie języku programowania **napisz program**, za pomocą którego uzyskasz odpowiedzi na poniższe polecenia. Odpowiedzi zapisz w pliku wyniki6.txt, a odpowiedź do każdego podpunktu poprzedź literą oznaczającą ten podpunkt.

a) Ile spośród liczb zapisanych w pliku dane.txt spełnia taki warunek, że pierwsza cyfra tej liczby jest równa ostatniej cyfrze tej samej liczby?

b) Podaj, ile liczb w pliku dane.txt, po ich zapisaniu w systemie dziesiętnym, spełnia warunek podany w podpunkcie a), tzn. pierwsza cyfra liczby jest równa ostatniej cyfrze tej samej liczby.

Przykład: Liczba 2038 = 13110 spełnia podany warunek, natomiast 71238 = 366710 nie spełnia żądanego warunku.

c) Ile spośród liczb zapisanych w pliku dane.txt spełnia warunek, że rozpoczynając od najbardziej znaczącej cyfry w zapisie ósemkowym, każda kolejna cyfra tej liczby nie jest mniejsza od cyfry poprzedzającej? Podaj największą i najmniejszą liczbę z pliku dane.txt, które spełniają ten warunek.

Przykład: Liczba 123357 spełnia podany warunek, ponieważ $1 \leq 2 \leq 3 \leq 3 \leq 5 \leq 7$.

Do oceny oddajesz plik(i) o nazwie(ach), zawierający(e) komputerową realizację Twojego rozwiązania, oraz plik tekstowy wyniki6.txt, zawierający odpowiedzi do podpunktów a, b i c (odpowiedź do każdego podpunktu poprzedź literą oznaczającą ten podpunkt).

Zadanie 6. Cyfry (10 pkt)

W kolejnych wierszach pliku cyfry.txt znajduje się 1000 liczb naturalnych, mniejszych niż 10^9 (jeden miliard), po jednej liczbie w każdym wierszu. **Napisz program**, który da odpowiedzi do poniższych podpunktów. Każdą odpowiedź zapisz w pliku zadanie4.txt, poprzedzając ją oznaczeniem odpowiedniego podpunktu.

a) Ile liczb parzystych jest w pliku cyfry.txt?

b) Podaj liczbę z pliku cyfry.txt, której suma cyfr jest **największa** oraz liczbę z tego pliku, której suma cyfr jest **najmniejsza**. W obu przypadkach jest tylko jedna taka liczba.

Przykład:

Dla danego zbioru liczb:

121324

66562

675100

1187010

odpowiedzią są liczby: **66562** oraz **121324**, ponieważ suma cyfr liczby 66562 jest równa 25 ($6+6+5+6+2$) i jest największą taką sumą, zaś suma cyfr liczby 121324 ($1+2+1+3+2+4$) jest równa 13 i jest najmniejszą taką sumą.

c) Wypisz wszystkie liczby z pliku cyfry.txt, których cyfry tworzą ciąg rosnący.

Przykład:

Cyfry liczby 123579 tworzą ciąg rosnący, ponieważ $1 < 2 < 3 < 5 < 7 < 9$.

Cyfry liczby 1232 nie tworzą ciągu rosnącego, ponieważ ostatnia cyfra (2) nie jest większa od przedostatniej (3).

Cyfry liczby 34556 nie tworzą ciągu rosnącego, ponieważ cyfra trzecia (5) i cyfra czwarta (5) są sobie równe.

Do oceny oddajesz plik zadanie4.txt oraz plik(i),

tu wpisz nazwę(y) pliku (ów)

zawierający(e) komputerową(e) realizację(e) Twojego rozwiązania.

Wskazówki do zadań

zad. 5

```
#include <iostream>
```

```
#include <cstdlib>
```

```
???? //biblioteka do obsługi plików
```

```
???? //biblioteka do obsługi plików funkcji działających na tekstach
```

```
using namespace std;
```

```
//funkcja sprawdzająca zwracająca w wyniku ile napisów ma parzystą długość
```

```
???? ile_parzystych ()
```

```
{   ???? //deklaracja zmiennej plikowej o nazwie odczyt służącej do czytania danych z pliku napisy.txt
```

```
    ???? //deklaracja tablicy tekstowej o nazwie napis do przechowywania napisów o długości conajwyżej 16 znaków
```

```
    ???? //definicja zmiennej całkowitej o nazwie ile z wartością początkową 0 do zliczania ile napisów jest parzystych
```

```
    ???? //deklaracja zmiennej całkowitej o nazwie dl do przechowywania informacji o długości wczatengo napisu
```

```
//pętla, która będzie wczytywała liczby z pliku, obliczała długość wczytanego napisu, sprawdzała czy ta długość jest podzielna przez 2 i ewentualnie zliczała ilość napisów o parzystej długości
```

```
    for (????)
```

```
    {   ???? }
```

```
odczyt.close();
```

```
return ile;
```

```
}
```

```
//funkcja sprawdzająca zwracająca w wyniku ile napisów ma taką samą ilość 1 i 0
```

```
???? rownosc ()
```

```
{   ???? //deklaracja zmiennej plikowej o nazwie odczyt służącej do czytania danych z pliku napisy.txt
```

```
    ???? //deklaracja tablicy tekstowej o nazwie napis do przechowywania napisów o długości conajwyżej 16 znaków
```

```
    ???? //definicja zmiennej całkowitej o nazwie ile z wartością początkową 0 do zliczania ile napisów gdzie liczba 0 i 1 w jednym napisie jest taka sama
```

```
    ???? //definicja zmiennych całkowitej o nazwie ile_0, ile_1 z wartością początkową 0 do zliczania ile w danym napisie jest 0 i ile jest 1
```

```
    ???? //deklaracja zmiennej całkowitej o nazwie dl do przechowywania informacji o długości wczytanego napisu
```

```
//pętla, która będzie wczytywała liczby z pliku, obliczała długość wczytanego napisu,
```

```
//a następnie w obrębie wczytanego napisu sprawdzała z ilu składa się 0 i z ilu 1
```

```
// na końcu będzie sprawdzała czy ilość zer jest taka sama czy różna i odpowiednio zwiększała wielkość zmiennej ile
```

```
for (????)
```

```
{   ile_1=0; //zastanów się po co taka instrukcja
```

```
    ile_0=0; //zastanów się po co taka instrukcja
```

```
    ????
```

```
}
```

```
odczyt.close();
```

```
return ile;
```

```
}
```

```
int takie_same (char z)
```

```
{   ifstream odczyt("napisy.txt");
```

```
    if (!odczyt)
```

```
        { cout<<"Pliku nie można otworzyć"; return 1;}
```

```
    else
```

```

    {
    char napis[17];
    int ile=0, dl;
    bool sa;
    for (int i=1;i<=1000;i++)
    { sa=true;
      odczyt>>napis;
      dl=strlen(napis);

      for (int j=0;j<dl;j++)
        if (napis[j]!=z) sa=false;
      if (sa==true) ile++;
    }
    odczyt.close();
    return ile;
  }

  }

void ile_jakich ()
{ ifstream odczyt("napisy.txt");
  if (!odczyt)
    { cout<<"Pliku nie można otworzyć"; }
  else
  { ofstream zapis("zadanie4.txt");
    int wynik[17];
    char napis[17];
    int ile=0, dl;
    bool sa=true;
    for(int j=0;j<17;j++) wynik[j]=0;

    for (int i=1;i<=1000;i++)
    { odczyt>>napis;
      dl=strlen(napis);
      wynik[dl]=wynik[dl]+1;
    }
    for(int j=2;j<17;j++)
      zapis<<j<<" znakowych jest "<<wynik[j]<<endl;
    odczyt.close();
    zapis.close();
  }
}

int main()
  {ofstream zapis("zadanie4.txt", ios::app);

  ile_jakich ();
  zapis<<"ilość napisow o parzystej dlugosci "<<ile_parzystych()<<endl;
  zapis<<"ilość napisow o rownej liczbie 1 i 0 to "<<rownosc()<<endl;
  zapis<<"ilość napisow gdzie same 0 "<< takie_same('0')<<endl;
  zapis<<"ilość napisow gdzie same 1 "<< takie_same('1')<<endl;
  cout<<"koniec";
  zapis.close();
  return 0;
  }
}

```

zad. 6

```
???? //dodaj potrzebne biblioteki
```

```
using namespace std;
```

```
//funkcja zwracająca w wyniku ile liczb jest parzystych
```

```
int parzyste()
```

```
{
```

```
    ???? //deklaracja zmiennej plikowej o nazwie odczyt służącej do czytania danych z pliku  
    cyfry.txt
```

```
    ???? /*deklaracja tablicy tekstowej o nazwie liczba do przechowywania liczb z pliku o  
    długości ??? zastanów się jakiej i dlaczego typ tekstowy */
```

```
    ???? //definicja zmiennej całkowitej o nazwie ile z wartością początkową 0 do zliczania ile  
    liczb jest parzystych
```

```
    ???? //deklaracja zmiennej całkowitej o nazwie dl do przechowywania informacji o  
    długości wczytanej liczby
```

```
    /*pętla, która będzie wczytywała liczby z pliku (pamiętajmy, że to teksty), obliczała ich  
    długość, sprawdzała i obliczała ile liczb jest parzystych, sprawdzała czy liczba jest parzysta  
    (np. można sprawdzać jaka jest ostatnia cyfra we wczytanej liczbie) i obliczała ile liczb jest  
    parzystych */
```

```
        for (????)
```

```
            { ???? }
```

```
        odczyt.close();
```

```
        return ??????? ;
```

```
    }
```

```
    /* funkcja która jako parametry zwróci w wyniku liczbę , której suma cyfr jest największa i  
    najmniejsza, & postawiony przed zmiennymi max i min powoduje, że wartość tych zmiennych  
    po wyjściu z funkcji zmieni się */
```

```
void suma_cyfr (string &max, string &min)
```

```
{
```

```
    ???? //deklaracja zmiennej plikowej o nazwie odczyt służącej do czytania danych z pliku  
    cyfry.txt
```

```
    ???? /*deklaracja tablicy tekstowej o nazwie liczba do przechowywania liczb z pliku o  
    długości ??? zastanów się jakiej i dlaczego typ tekstowy*/
```

```
    ???? //deklaracja zmiennej całkowitej o nazwie dl do przechowywania informacji o  
    długości wczytanej liczby
```

```
    ???? /*definicja zmiennej całkowitej o nazwie suma_max, do przechowywania  
    największej sumy cyfr wczytanej liczby z wartością początkową ???? jaka pomyśl  
    sam?*/
```

```
    ???? /*definicja zmiennej całkowitej o nazwie suma_min, do przechowywania  
    najmniejszej sumy cyfr wczytanej liczby z wartością początkową ???? jaka pomyśl  
    sam?*/
```

```

???? /*deklaracja tablicy jednowymiarowej o nazwie wartosc_cyfry, w której zapiszesz
kody ASCII cyfr odpowiadających numerowi indeksu tablicy, jaki będzie miała
wymiar tablica?*/

???? //deklaracja zmiennej całkowitej wartosc, w której znajdzie się suma cyfr wczytanej
liczby

/*pętla ma za zadanie tablicy wartosc_cyfry przypisać odpowiednią wartości kodów ASCII,
np. dla indeksu 0 tablicy ma wpisać liczbę 48, dla indeksu 1 liczbę 48+1 itd.*/
for (????) ???

/*pętla, która będzie wczytywała liczby z pliku (pamiętajmy, że to teksty), obliczała ich
długość dla wczytanej liczby, obliczała jaka suma jej cyfr i wybierała liczbę, której suma cyfr
jest największa i najmniejsza */
for (???????)
{
    ???????

    /*pętla, która dla danej liczby (która w pliku traktowana jest jako tekst), dla każdego
pojedynczego znaku wyznacza jego wartość porównując kod ASCII tego znaku z kodem
ASCII w tablicy wartosc_cyfry */
    for(int j=0;j<dl;j++)
        for (int k=0;k<10;k++)
            if( (int) liczba[j] == wartosc_cyfry[k] ) wartosc = wartosc + k;

    /*instrukcja warunkowa która sprawdza czy bieżąco obliczona wartość jest większa od
dotychczasowej wartości max*/
    if (wartosc>suma_max) {suma_max=wartosc; max=liczba; }

    /*instrukcja warunkowa która sprawdza czy bieżąco obliczona wartość jest mniejsza od
dotychczasowej wartości min*/
    if (wartosc<suma_min) {suma_min=wartosc; min=liczba; }
}
odczyt.close();
}

/*funkcja, która zwraca wartość prawda, gdy argument będący tablicą tekstową składa się z
cyfr tworzących ciąg rosnący*/
???? ciag_rosnacy (char liczba[])
{
    ????? //deklaracja zmiennej całkowitej o nazwie dl do przechowywania informacji o
długości wczytanej wartości

    ????? /*definicja zmiennej logicznej o nazwie ok z wartością początkową true, która
decyduje czy cyfry tworzą ciąg rosnący czy nie, na początku zakładamy że tworzą
*/

    ????? //instrukcja przypisująca zmiennej dl długość wczytanej wartości

/*pętla, która się zakończy jeśli cyfra wcześniejsza będzie większa bądź równa od cyfry
późniejszej

```


*jeśli taka sytuacja nie będzie miała miejsca to oznacza, że cyfry w liczbie tworzą ciąg rosnący a wartość zmiennej ok=true */*

```
for (???????)  
    if (?????) {ok=false;i=dl;}
```

```
return ok;  
}
```

//funkcja która wyświetli liczby z pliku cyfry.txt, których cyfry tworzą ciąg rosnący

```
void wyswietl_ciag_rosnacy()
```

```
{  
    ???? //deklaracja zmiennej plikowej o nazwie odczyt służącej do czytania danych z pliku cyfry.txt
```

```
    ???? /*deklaracja tablicy tekstowej o nazwie liczba do przechowywania liczb z pliku o długości ??? zastanów się jakiej i dlaczego typ tekstowy*/
```

*/*pętla, która odczyta liczbę z pliku cyfry.txt sprawdzi wykorzystując funkcję ciag_rosnacy czy spełnia określony warunek a jeśli tak to ją wyświetli*/*

```
for (???????)  
{  
    ??????  
    odczyt.close();  
}
```

//w głównej funkcji należy wywołać, funkcje napisane zgodnie z podpunktami

```
int main()
```

```
{ //wywołanie funkcji realizującej podpunkt a) czyli parzyste()
```

//deklaracja zmiennych typu string max, min, które będą potrzebne od wywołania kolejnej funkcji

```
//wywołanie funkcji realizującej podpunkt b) czyli suma_cyfr(max,min)
```

```
//wyświetlenie zmiennych max i min
```

```
//wywołanie funkcji realizującej podpunkt c) wyswietl_ciag_rosnacy()
```

```
cout<<"koniec";
```

```
return 0;
```

```
}
```

wyniki działania programu:

a) parzystych liczb jest 495

b) max= 187869866 min= 10010

c) 2389 ,23567, 123456789, 3468

Rozwiązanie zad. 5

```
#include <iostream>
#include <fstream>
using namespace std;

//realizacja a) ile liczb w pliku, takich że pierwsza cyfra = ostatniej cyfrze
int ile(char nazwa[])
{ ifstream dane(nazwa);

  string liczba;
  int rozmiar;
  int licz=0;
  for (int i=1;i<=5000;i++)
    {dane>>liczba;
      rozmiar=liczba.size();

      if (liczba[0]==liczba[rozmiar-1]) licz++;
    }
  return licz;
}

void zamiana(char nazwa1[],char nazwa2[])
{ ifstream dane(nazwa1);
  ofstream wynik1(nazwa2);

  int cyfry[2][10];

  for (int i=0;i<10;i++) { cyfry[0][i]=i;   cyfry[1][i]=48+i; } //przypisanie cyfromo od 0 do 9 kodów ASCII od 48

  string liczba1,liczba2;
  int rozmiar;
  int wartosc;

  for (int i=1;i<=5000;i++)
    { dane>>liczba1;
      rozmiar=liczba1.size();

      int p8=1;
      int wartosc=0;
      int w;

      for (int j=rozmiar-1;j>=0;j--)
        { for (int k=0;k<10;k++)
            if((int)liczba1[j]==cyfry[1][k]) w=cyfry[0][k];
            wartosc=wartosc+w*p8;
            p8=p8*8;
          }
        wynik1<<wartosc<<endl;
      }
  ile(nazwa2);
}

int mniejsze(char nazwa[])
{ ifstream dane(nazwa);

  int cyfry[2][10];

  for (int i=0;i<10;i++) { cyfry[0][i]=i;   cyfry[1][i]=48+i; }
```

```

string liczba;
int rozmiar;
int licz=0;
int w1,w2;

for (int i=1;i<=5000;i++)
{
    dane>>liczba;
    rozmiar=liczba.size();

    for (int k=0;k<10;k++)
        if((int)liczba[0]==cyfry[1][k]) w1=cyfry[0][k];

    int j=1;
    for (j=1;j<rozmiar;j++)
    {
        for (int k=0;k<10;k++)
            if((int)liczba[j]==cyfry[1][k]) w2=cyfry[0][k];

        if (w1<=w2) w1=w2;
        else j=rozmiar;
    }
    if (j==rozmiar) licz++;
}
return licz;
}

int max(char nazwa1[])
{ ifstream dane(nazwa1);

int cyfry[2][10];

for (int i=0;i<10;i++) { cyfry[0][i]=i; cyfry[1][i]=48+i; }

string liczba1,liczba2;
int rozmiar;
int wartosc;
int Max=-1;
for (int i=1;i<=5000;i++)
{ dane>>liczba1;
rozmiar=liczba1.size();

int p8=1;
int wartosc=0;
int w;

for (int j=rozmiar-1;j>=0;j--)
{ for (int k=0;k<10;k++)
    if((int)liczba1[j]==cyfry[1][k]) w=cyfry[0][k];
    wartosc=wartosc+w*p8;
    p8=p8*8;
}
if(Max<wartosc) Max=wartosc;
}
}

```

```

return Max;
}
int min(char nazwa[])
{ ifstream dane(nazwa);

int cyfry[2][10];

for (int i=0;i<10;i++) { cyfry[0][i]=i; cyfry[1][i]=48+i; }

string liczba;
int rozmiar;
int wartosc;
int Min;
Min=max(nazwa);
for (int i=1;i<=5000;i++)
{ dane>>liczba;
rozmiar=liczba.size();

int p8=1;
int wartosc=0;
int w;

for (int j=rozmiar-1;j>=0;j--)
{ for (int k=0;k<10;k++)
if((int)liczba[j]==cyfry[1][k]) w=cyfry[0][k];
wartosc=wartosc+w*p8;
p8=p8*8;
}
if(Min>wartosc) Min=wartosc;
}
return Min;
}
int main()
{ cout<<"a) "<<ile("dane.txt")<<endl;
cout<<"b) "<<ile("wynik1.txt")<<endl;
cout<<"c) "<<mniejsze("dane.txt")<<endl;
cout<<"max = "<<max("dane.txt")<<endl;
cout<<"min = "<<min("dane.txt")<<endl;
//zamiana();
cin.get();
getchar();
return 0;
}

```