***Злиття двох відсортованих файлів*  *Обчислити інтеграл з межами***

#include <iostream> ***double gorner***(double\*A, int n, double x)

#include <fstream> {double Sum = A[0];

using namespace std; for (int i = 0; i < n; i++)

***void main***(){ {Sum = Sum\*x + A[i + 1];}

ifstream f("f.txt"); return Sum;}

ifstream g("g.txt"); ***void main***(){

ofstream ff("ff.txt"); int n; double c, d;

double n1, n2; cout << "Power of polinom" << endl;

if (f.is\_open() && g.is\_open()) { cin >> n;

f >> n1; g >> n2; double\*A = new double[n + 2];

while (!f.eof() && !g.eof()) cout << "Input limits of integral, c & d";

{ if(f.eof()) cin >> c >> d;

while (!g.eof()) { g >> n2; ff << n2; } cout << "Input coefficients of polinom" ;

if (g.eof()) for (int i = 0; i < n+1; i++){

while (!f.eof()) { f >> n1; ff << n1; } cout << "A[" << i << "]= ";

if (n1 > n2) { ff << n2; g >> n2; } cin >> A[i];

if (n1 < n2) { ff << n1; f >> n1; } A[i] /= n + 1 - i;}

if (n1 == n2) { ff << n1; f >> n1; g >> n2; } A[n + 1] = 0;

} cout << gorner(A, n + 1, d) - gorner(A, n + 1, c);

} }

else cout << " Can't open file! " << endl;

system("pause");

}

***Стек з числа*** ***Кількість слів у стрічці(Version 1.0)***

#include <iostream> #include <iostream>

using namespace std; using namespace std;

***struct stek*{ *void main*(){**

int ch; char str[250];

stek \*next;**};** int count = 1;

***stek StekCreating*(long N){** FILE \*stream;

stek \*p = new stek; int err;

p->next = NULL; err = fopen\_s(&stream, "test.txt", "r");

p->ch = N % 10; N /= 10; if (err == 0){

while (N) fgets(str, 250, stream);

{ for (int i = 0; i < strlen(str); i++)

stek \*k = new stek; { if (str[i] == ' ')

k->next = &\*p; { count++; }

k->ch = N % 10; N /= 10; }

p = k; cout << "Words count = " << count << "\n";

} system("pause ");}

return \*p;**} }**

***void main***(){

long N;

cin >> N; ***Для заданого Р обчислити похідні Р'(2.34),Р'(7.2),Р'(5.3).***

stek \*i = new stek; #include<iostream>

\*i=StekCreating(N); using namespace std;

while (i){ ***void main***(){

cout << i->ch<<" "; int n; double x;

i = i->next;} cout << "Please, input power of polinom";

system("pause");} cin >> n;

***Координати min max у файлі*** cout << "Please input X " << endl;

#include<iostream> cin >> x;

#include <fstream> cout << "Please input coeficients" << endl;

using namespace std; double \*Coef = new double[n];

***void main***() for (int i = 0; i < n; i++)

{ {cin >> Coef[i];}

ifstream f("text.txt"); double F = Coef[0]\*(n - 1);

int min=0, max=0; for (int i = 1; i < n - 1; ++i)

double curpos, minf, maxf; {F = F\*(n - i - 1)\*x + Coef[i];}

f >> curpos; minf = maxf = curpos; int i = 1; cout << F << endl;

while (!f.eof()) }

{

f >> curpos;

if (curpos > maxf) { maxf = curpos; max=i; }

if (curpos < minf) { minf = curpos; min = i;}

i++;

}

max++; min++;

cout << max << " " << min << endl;

system("pause");

}

***Уточнити корінь многочлена(ч.1) Уточнити корінь многочлена(ч.2)***

***int count\_digits***(ifstream &file) ***void main***(){

{ ifstream file; int n;

int counter = 0; double eps=0, x0=0,x1;

double digit; file.open("file.txt");

while (file >> digit) if (!file.is\_open())

{counter++;} {cout << "File not found!" << endl;}

file.clear(); cout << "input eps\t";

file.seekg(0); cin >> eps;

return counter; cout << "input x0\t";

} cin >> x0; x1 = x0;

***double polinom\_value***(ifstream &file,double x,int n) n = count\_digits(file);

{ do {

double a; x0 = x1;

file >> a; x1=x0-polinom\_value(file, x0, n)/polinom\_derive(**\***);

double F = a; } while (abs(x0 - x1) >= eps);

for (int i = 1; i < n; ++i) cout << "root of equation\t" << x1;

{file >> a;F = F\*x + a;} system("pause>>void");

file.clear(); }

file.seekg(0); Примітка: **\* передаємо те саме в другу функ-цію**

return F;

}

***double polinom\_derive***(ifstream &file,double x,int n)

{

double a;

file >> a;

double F = a\*(n-1);

for (int i = 1; i < n-1; ++i)

{file >> a;F = F\*(n-i-1)\*x + a;}

file.clear();

file.seekg(0);

return F;

}

***A/B(різниця двох списків)*** ***Обчислення інтегралу***

#include <iostream> #include <iostream>

#include <vector> using namespace std;

#include <iomanip> void main(){

using namespace std; double x; int n;

***struct stek***{ cout << "please input n" << endl;

int ch; cin >> n;

stek \*next; cout << "input x";

}\*A,\*B; cin >> x;

***void push***(stek \*&p, int numb) double \*A = new double[n + 1];

{stek \*k = new stek; for (int i = 0; i <= n;i++)

k->next = p; {cout << "A[" << i << "]=\t";

k->ch = numb; cin >> A[i];}

p = k;} double P=n\*A[0];

***void main***() for (int i = 1; i <= n - 1;i++)

{double helpnumb; bool b = 1; {P = A[i] \* (n - i) + x\*P;}

srand(time(0)); cout << P;

A = new stek; A = NULL; system(“pause”);

B = new stek; B = NULL; }

for (int i = 0; i < 10;i++) ***Кількість слів у стрічці(Version 2.0)***

{push(A, rand() % 15); #include <iostream>

push(B, rand() % 10);} #include <fstream>

stek \*q = new stek; q = A; using namespace std;

cout << "A : "; ***void main***()

while (q) {const int maxword = 1024;

{cout << q->ch<<" "; char word[maxword];

q = q->next;}cout << endl; ifstream file;

q = B; cout << "B : "; file.open("file.txt");

while (q) int counter(0);

{cout << q->ch<<" "; if (!file.is\_open())

q = q->next;}cout << endl; {cout << "File si not found!" << endl;}

cout << "A/B : "; while (file >> word)

while (A){q = B; {strtok(word,",./n /t1234567890");

while (q) for (int i = 0; i < strlen(word); i++)

{if (A->ch == q->ch) {if (isalpha(word[i]))

{b = 0; break;} {counter++;

else { b = 1; helpnumb = A->ch; } break;}}}

q = q->next; file.close();

}if (b)cout << helpnumb<<" "; cout <<"In file "<<counter<<" word(s)";

A = A->next;} system("pause>>void");}

}

***Перетин файлів і Злиття (\*)*** ***Функція переводу числового файлу в бінарний (\*\*)***

#include <iostream> ***void intsToBin(***ifstream&fInTxt, fstream&foutBin)

#include <fstream> {fInTxt.seekg(0,ios::beg);

using namespace std; foutBin.seekg(0,ios::beg);

***void main*** (){ int temp;

ifstream f, g; while(!fInTxt.eof())

ofstream ff; {fInTxt >> temp;

f.open("f.dat",ios\_base::in|ios::binary); fOutBin.write((char\*)&temp, sizeof(int));}

g.open("g.dat", ios\_base::in|ios::binary); fOutBin.flush();

ff.open("ff.dat", ios\_base::out|ios::binary); }

if (!f.is\_open() || !g.is\_open())

{cout << "File not found!" << endl; ***Злиття списків***

return 1;} ***double \*smerge*** (double \*s, іnt m, іnt n)

if (!ff.is\_open()) { іnt l,low,up;

{cout << "File not created!" << endl; double \*merge (double \*, іnt, іnt, іnt);

return 1;} l=1;

while (!f.eof()) whіle(l<=(n-m))

{double f\_tmp=0; { low=m;

f.read((char\*)&f\_tmp, sizeof(f\_tmp)); up=m-1;

while (!g.eof()) whіle (l+up < n)

{double g\_tmp=0; { up=(low+2\*l-1 < n) ? (low+2\*l-1) : n ;

g.read((char\*)&g\_tmp, sizeof(g\_tmp)); merge (s,low,up,l);

if (g.eof()) low=up-1;}

break; l\*=2;}

cout <<'g'<< g\_tmp << endl; return(a);}

if (f\_tmp == g\_tmp ***(\*)***

{ff.write((char\*)&g\_tmp, sizeof(g\_tmp)); ***/\* рекурсивне сортування злиттям 1/2 \*/***

cout << '\t';break;} } ***double \*srecmg*** (double \*a, іnt m, іnt n)

if(g.eof()) g.clear(); { double \* merge (double \*, іnt, іnt, іnt);

g.seekg(0, ios\_base::beg|ios::binary); double \* smerge (double \*, іnt, іnt);

} іnt і;

f.close();g.close();ff.close(); іf (n>m)

} { і=(n+m)/2;

***(\*)***Тут якщо поміняти на(!=)то вже буде злиття srecmg(a,m,і);

***(\*\*)***Якщо файл числовий, використати функцію srecmg(a,і+1,n);

***Обчислити коефіцієнти Н100*** merge(a,m,n,(n-m)/2+1);}

#include <iostream> return(a);}

using namespace std;

int main(){ ***Сортування мін та макс елем***  ***double \*select***( double \*s, іnt m, іnt n)

double a, b; { іnt і,i\_max,i\_min; double c;

int n; while (m<n)

cout << " input a,b\t"; { i\_max=i\_min=m;

cin >> a >> b; for (і=m+1; і<=n; i++)

double \*H0 = new double[1]; іf(s[і] > s[i\_max]) i\_max =і;

H0[0] = 5; else іf(s[і] < s[i\_min]) i\_min=i;

double \*H1 = new double[2]; /\* обмін \*/

H1[0] = b; іf(n!=i\_max) {c=s[n]; s[n--]= s[i\_max];

H1[1] = a; s[i\_max]= c;}

cout << "input n\t"; іf(m!= i\_min) {c=s[m]; s[m++]= s[i\_min];

cin >> n; s[i\_min]= c;}}

for (int i = 2; i <= n; i++) return(s);}

{

double \*H2 = new double[i + 1];

for (int j = 0; j < i + 1; j++)

{H2[j] = 0;}

for (int j = 1; j < i + 1; j++)

{H2[j] = H1[j - 1];}

for (int j = 0; j < i - 1; j++)

{H2[j] += H0[j];}

delete[]H0;

H0 = H1;

H1 = H2;}

for (int i = n; i >= 0; i--)

{if(H1[i]!=0)

{if(i>0)

cout << H1[i] << "\*x^"<<i<<"+";

else

cout << H1[i] <<endl;}}

system("pause>>void");}