**Код программы:**

#include <stdio.h>

int StringLenght(char \*s);

void BinaryFileRead(const char \*filename);

const char \*filename1 = "input.txt";

const char \*filename2 = "output.lol";

int main()

{

char s[255];

FILE \*in;

FILE \*on;

double CountOfLines;

int Length;

int Maxlength;

in = fopen(filename1, "rt");

//Если файл открылся:

if (in != NULL)

{

CountOfLines = 0;

Maxlength = 0;

while (fgets(s, 256, in) != NULL)

{

//Подсчет кол-ва символов в StringLenght()

Length = StringLenght(s);

CountOfLines++;

if (Length > Maxlength)

Maxlength = Length;

printf("%s", s);

}

if (!feof(in))//Если не конец файла

{

fclose(in);

printf("error of reading\n");

getchar();

return 0;

}

printf("\n");

printf("Size of lenghts %.0f\n", CountOfLines);

if(fseek(in, 0, SEEK\_SET) != 0)

{

printf("Error of programm");

fclose(in);

return -1;

}

on = fopen(filename2, "wb");

if (on != NULL)

{

printf("Recording of binary file\n\n");

if (fwrite(&CountOfLines, sizeof(double), 1, on) != 1)

{

fclose(on);

fclose(in);

remove(filename2);

printf("error of writing\n");

return 0;

}

while (fgets(s, 255, in) != NULL)

{

//Если запись в бинарный файл прошла некорректно

if (fwrite(s, sizeof(char), Maxlength, on) !=

Maxlength)

{

fclose(on);

fclose(in);

remove(filename2);

printf("error of writing\n");

return 0;

}

}

fclose(on);

fclose(in);

printf("End of recording of binary file\n");

BinaryFileRead(filename2);

}

else

{

fclose(in);

printf("file can't be created or found\n");

getchar();

return 0;

}

}// end of "if (in != NULL)"

else

{

printf("file not found\n");

getchar();

return 0;

}

getchar();

return 0;

}

int StringLenght(char \*s)

{

int i = 0;

while ((s[i] != NULL) && (s[i] != '\n') && (s[i] != 0))

i++;

return i;

}

void BinaryFileRead(const char \*filename)

{

char l[255] = { 0 };

FILE \*bn;

double CountOfLines;

long Size;

int i;

int RealSizeOfLenghts;

int SizeOfLenghts;

bn = fopen(filename, "rb");

if (bn != NULL)

{

if(fseek(bn, 0, SEEK\_END) != 0)

{

printf("Error of displacement in binary file");

fclose(bn);

return;

}

Size = ftell(bn); //Определение размера файла

if(fseek(bn, 0, SEEK\_SET) != 0)

{

printf("Error of displacement in binary file");

fclose(bn);

return;

}

if (Size <= 0)

{

fclose(bn);

printf("size of file isn't correct\n");

return;

}

//Неудачное считывание бинарного файла

if (fread(&CountOfLines, sizeof(double), 1, bn) != 1)

{

fclose(bn);

printf("error of reading\n");

return;

}

else

{

if (CountOfLines <= 0)

{

printf("File is incorrect");

fclose(bn);

return;

}

else

{

printf("Size of bin file is %ld byte(s)\n", Size);

}

//sizeof(double) - размер значения длины строки

SizeOfLenghts = (Size - sizeof(double))/CountOfLines;

for (i = 0; i < (int)CountOfLines; i++)

{

if(fread(l, sizeof(char), SizeOfLenghts, bn) !=

SizeOfLenghts)

{

printf("I cant read %d values\n", SizeOfLenghts);

break;

}

else

{

RealSizeOfLenghts = StringLenght(l);

printf("RealSizeOfLenghts %d " ,

RealSizeOfLenghts);

//Указание конца строки

l[RealSizeOfLenghts] = '\0';

printf("%s\n", l);

}

}

fclose(bn);

}

}

else

{

printf("file not found\n");

return;

}

}