Файл PackUnpack.h

#pragma once

#include <stdio.h>

#include <windows.h>

#include <conio.h>

#define STRING\_SIZE 100

#define MENU\_SMALL\_SIZE 3

#define MENU\_X 5

#define MENU\_Y 5

static char menu[MENU\_SMALL\_SIZE][24] = { "ВВОД ЦЕЛЫХ ЧИСЕЛ",

"ВВОД ВЕЩЕСТВЕННЫХ ЧИСЕЛ",

"ВЫВОД ВВЕДЕННЫХ ЧИСЕЛ" };

#define MENU\_POS\_1 0

#define MENU\_POS\_2 1

#define MENU\_POS\_3 2

#define ESC 27

#define ENTER 13

#define UP 72

#define DOWN 80

enum colors\_enum

{

Black = 0,

White = FOREGROUND\_RED | FOREGROUND\_GREEN | FOREGROUND\_BLUE | FOREGROUND\_INTENSITY,

LightPurple = FOREGROUND\_RED | FOREGROUND\_BLUE | FOREGROUND\_INTENSITY

};

void gotoxy(int x, int y);

void cursorhide();

void setcolor(WORD attr);

int MyPacking(char\* pointer);

void MyUnpacking(char\* pointer);

int SizeControl(int data\_size, int next\_num);

HANDLE hStdOut;

Файл Main.c

#include "PackUnpack.h"

int main()

{

system("chcp 1251");

system("cls");

hStdOut = GetStdHandle(STD\_OUTPUT\_HANDLE);

cursorhide();

char string[STRING\_SIZE];

int control = MyPacking(string);

if (control == 1) return 1;

MyUnpacking(string);

return 0;

}

Файл MyPacking.c

#include "PackUnpack.h"

int MyPacking(char\* pointer) {

int i, pos = 0;

int isExit = 0;

char ch;

char\* pointer\_copy = pointer;

int numbers;

system("cls");

while (!isExit)

{

for (i = 0; i < MENU\_SMALL\_SIZE; i++)

{

gotoxy(MENU\_X, MENU\_Y + i);

if (i == pos)

{

setcolor(LightPurple);

}

else

{

setcolor(White);

}

printf("%s", menu[i]);

}

ch = getch();

switch (ch)

{

case ESC:isExit = 1; break;

case UP:

if (pos > 0)

{

pos--;

}

break;

case DOWN:

if (pos < MENU\_SMALL\_SIZE - 1)

{

pos++;

}

break;

case ENTER:

switch (pos)

{

case MENU\_POS\_1: system("cls");

printf("Введите желаемое кол-во целых чисел: ");

scanf("%d", &numbers);

\*pointer++ = '%'; // кодируем начало записи

if (numbers > 1)

\*pointer++ = (char)numbers; // размер массива, если массив

\*pointer++ = 'd'; // кодируем конец записи

int integer\_int;

for (int j = 0; j < numbers; j++) {

printf("Введите ваше %d-е целое число: ", j + 1);

scanf("%d", &integer\_int);

isExit = SizeControl(pointer - pointer\_copy, integer\_int); // контроль переполнения массива

if (isExit == 1) return 1;

\*((int\*)pointer)++ = integer\_int; // сохраняем введенные числа после записи

}

printf("\n");

system("cls");

break;

case MENU\_POS\_2: system("cls");

printf("Введите желаемое кол-во вещественных чисел: ");

scanf("%d", &numbers);

\*pointer++ = '%'; // кодируем начало записи

if (numbers > 1)

\*pointer++ = (char)numbers; // размер массива, если массив

\*pointer++ = 'f'; // кодируем конец записи

float float\_f;

for (int k = 0; k < numbers; k++) {

printf("Введите ваше %d-е вещественное число: ", k + 1);

scanf("%f", &float\_f);

isExit = SizeControl(pointer - pointer\_copy, (int)float\_f); // контроль переполнения массива

if (isExit == 1) return 1;

\*((float\*)pointer)++ = float\_f; // сохраняем введенные числа после записи

}

printf("\n");

system("cls");

break;

case MENU\_POS\_3: isExit = 1; break;

}

break;

}

}

gotoxy(0, 0);

setcolor(White);

return 0;

}

Файл MyUnpacking.c

#include "PackUnpack.h"

void MyUnpacking(char\* pointer) {

char\* ptr\_cpy = pointer;

int numbers;

int i;

system("cls");

while (\*pointer++ == '%') {

if (\*pointer == 'd' || \*pointer == 'f') { // проверка: одно число или массив

numbers = 1;

}

else {

numbers = (int)\*pointer++; // берем размер массива

}

switch (\*pointer++) {

case 'd': printf("Было введено целых чисел: %d, со значениями: ", numbers);

int integer\_int;

for (i = 0; i < numbers; i++) {

integer\_int = \*((int\*)pointer)++;

printf("%d ", integer\_int);

}

printf("\n");

break;

case 'f': printf("Было введено вещественных чисел: %d, со значениями: ", numbers);

float float\_f;

for (i = 0; i < numbers; i++) {

float\_f = \*((float\*)pointer)++;

printf("%.2f ", float\_f);

}

printf("\n");

break;

}

}

getch();

}

Файл SizeControl.c

#include "PackUnpack.h"

int SizeControl(int data\_size, int next\_num) {

// контроль переполнения массива

if (next\_num > 9999 || data\_size + sizeof(int) > STRING\_SIZE - 1) {

printf("Переполнение памяти! Пожалуйста, попробуйте еще раз.");

getch();

return 1;

}

return 0;

}