Московский Авиационный Институт

(Национальный Исследовательский Университет)

Кафедра: 806 «Вычислительная математика и программирование»

Факультет: «Прикладная математика и физика»

Дисциплина: «Операционные системы»

Лабораторная работа №5.

Группа: 8О-204Б

Студент: Лебедев Т.А.

Вариант: №

Оценка:

Подпись:

Москва

2016

#include <stdlib.h>

#include <stdio.h>

#include "array.cpp"

int main()

{

int a;

dummy();

scanf("%d", &a);

return 0;

}

#define Array\_TYPE char

#include <stdlib.h>

#include <stdio.h>

typedef struct \_ARRAY

{

int \_size;

Array\_TYPE Data[n][32];

} Array;

\_\_declspec(dllexport)

void arrayPrint(Array \*s)

{

int i, j;

for (i = 0;i<s->\_size;i++)

{

for (j = 0;j<32;j++)

{

printf("%c", s->Data[i][j]);

}

printf("\n");

}

printf("\n");

}

\_\_declspec(dllexport)

void arrayCreate(Array \*s)

{

s->\_size = 0;

}

\_\_declspec(dllexport)

void arraySize(const Array \*s)

{

printf("%d\n", s->\_size);

}

\_\_declspec(dllexport)

int arrayInsert(Array \*s, Array\_TYPE value[32])

{

int j;

int i = s->\_size;

for (j = 0;j<32;j++)

{

s->Data[i][j] = value[j];

}

s->\_size++;

return 1;

}

\_\_declspec(dllexport)

int arrayDeleteElem(Array \*s, Array\_TYPE value[32])

{

int i, j;

if (!s->\_size)

{

return 0;

}

for (i = 0;i<(s->\_size);i++)

{

for (j = 0;j<32;j++)

if (s->Data[i][j] == value[j])

{

while (i != s->\_size)

{

for (j = 0;j<32;j++)

s->Data[i][j] = s->Data[i + 1][j];

i++;

}

s->\_size--;

return 1;

}

}

}

\_\_declspec(dllexport)

void arrayDestroy(Array \*s)

{

int i, j;

for (i = 0;i<(s->\_size);i++)

{

for (j = 0;j<32;j++)

{

(s->Data[i][j]) = NULL;

}

}

s->\_size = 0;

printf("delete!\n");

}

extern "C" \_\_declspec(dllexport) void dummy()

{

Array p;

arrayCreate(&p);

arrayInsert(&p, "caf1a3dfb505ffed0d024130f58c5cfa");

arrayInsert(&p, "1BC29B36F623BA82AAF6724FD3B16718");

arrayInsert(&p, "9b04d152845ec0a378394003c96da594");

arrayInsert(&p, "0802ba9cb8ff96a671e3ecd56f2d21e4");

arrayPrint(&p);

arraySize(&p);

arrayDeleteElem(&p, "1BC29B36F623BA82AAF6724FD3B16718");

arrayPrint(&p);

arraySize(&p);

arrayDestroy(&p);

arrayPrint(&p);

arraySize(&p);

}

#include <Windows.h>

#include <stdio.h>

#include <conio.h>

typedef void (\*function)(void);

int main()

{

HINSTANCE hm = LoadLibrary(L"0s5fin.dll");

if (hm == NULL)

{

printf("DLLvscode.dll not found!\n");

}

else

{

function dummy = (function)GetProcAddress(hm, "dummy");

if ((dummy == NULL) )

{

printf("DLLvscode.dll doesn't export add or sub\n");

}

else

{

dummy();

}

}

\_getch();

FreeLibrary(hm);

return 0;

}

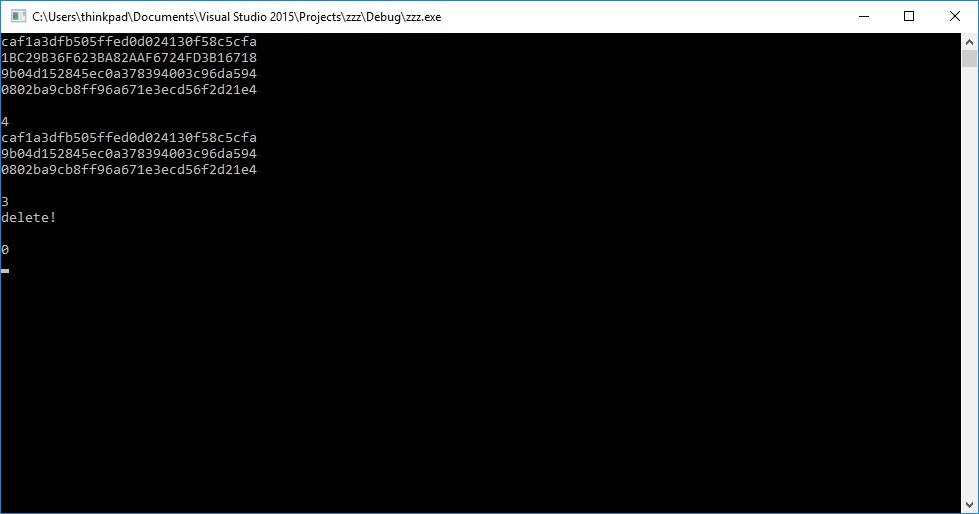
def-file

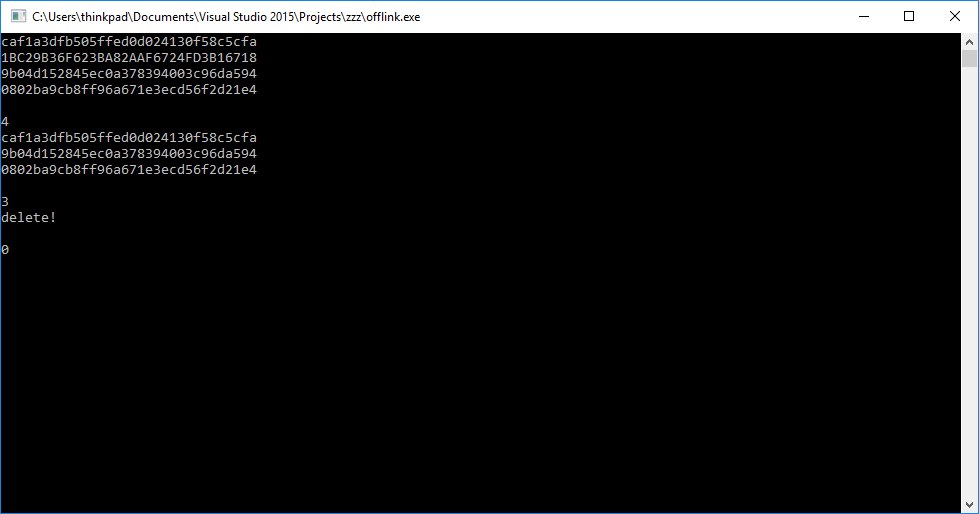
LIBRARY "0s5fin"

EXPORTS

dummy

dead





Вывод

Особой разницы при выполнении программы при различном подключении библиотек нет.

Но динамический способ позволит нам немного сохранить память при выполнении, так как нам не нужно сразу загружать все необходимые библиотеки в оперативку, а подключать по мере необходимости и отключать, если возникнут проблемы с нехваткой памяти.