ForNextDay(18)  
Stephen Cole

3553803

strinlisttest.c

#include"StringList.h"

#include"Strings.h"

#include<stdlib.h>

#include<stdio.h>

int main(int argc, char\*\* argv)

{

StringList\* sl = mallocStringList(argc);

printf("Original\n");

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

{

printf("%s\n", argv[i]);

setElementStringList(sl,argv[i],i);

}

StringList\* slDup;

slDup = duplicateStringList(sl);

printf("Duplicate\n");

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

printf("%s\n", getElementStringList(sl,i));

return 0;

}

output

A picture containing flower

Description automatically generated

StringList.c

StringList\* mallocStringList(int iNElements)

{

StringList\* pSLThis = (StringList\*)malloc(sizeof(StringList));

pSLThis->Slist = (String\*)malloc(iNElements \* sizeof(String));

pSLThis->length = iNElements;

for (int i = 0; i < pSLThis->length; i++) {

pSLThis->Slist[i] = (String)NULL;

}

return pSLThis;

}

void freeStringList(StringList\* pSLThis)

{

for (int i = 0; i < pSLThis->length; i++) {

free(pSLThis->Slist[i]);

}

free(pSLThis);

}

int setElementStringList(StringList\* pSLThis, String sThis, int index)

{

// How do we tell if the element previously held a String?

if (pSLThis->Slist[index] != (String)NULL) {

freeString(pSLThis->Slist[index]);

}

String s = duplicateString(sThis);

if (s == (String)NULL)

return -1;

pSLThis->Slist[index] = s;

return index;

}

String getElementStringList(StringList\* pSLThis, int index)

{

if (pSLThis->Slist[index] == (String)NULL)

return (String)NULL;

return duplicateString(pSLThis->Slist[index]);

}

StringList\* duplicateStringList(StringList\* pSLThis)

{

StringList\* pSLThat = mallocStringList(pSLThis->length);

if (pSLThat == (StringList\*)NULL)

return pSLThat;

for (int i = 0; i < pSLThis->length; i++) {

String str = getElementStringList(pSLThis, i);

int index = setElementStringList(pSLThat, str, i);

if (index == -1) {

freeStringList(pSLThat);

return (StringList\*)NULL;

}

}

return pSLThat;

}