Multiple Strings and Sorting with an Array of Pointers

Supplementary Slides to Lecture 7

Yakup Genc

```
#define NAMELENGTH 8
#define NUMFLOWERS 5
void test() {
    char flowers[NUMFLOWERS][NAMELENGTH] =
        {"tulip", "marigld", "petunia", "rose", "daisy"};
    char * index[NUMFLOWERS];
    char * t;
    int i, j;
    /* Set up addresses in the array to point to actual strings */
    for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre>
    for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);
    for (i=0; i<NUMFLOWERS; i++) {</pre>
        for (j=i+1; j<NUMFLOWERS; j++) {</pre>
            if (strcmp(index[i], index[j]) > 0) {
                t = index[i];
                index[i] = index[j];
                index[j] = t;
    for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);
```

Before sorting:

FFFFCBAO: tulip

FFFFCBAA: marigld

FFFFCBBE: rose

FFFFCBC8: daisy

After sorting:

FFFFCBC8: daisy

FFFFCBAA: marigld

FFFFCBBA: petunia

FFFFCBBE: rose

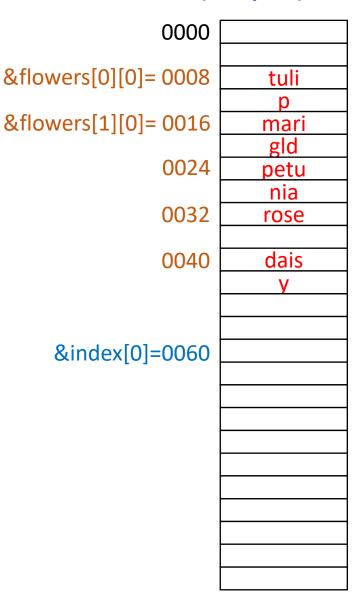
FFFFCBA0: tulip

```
#define NAMELENGTH 8
#define NUMFLOWERS 5
void test() {
    char flowers[NUMFLOWERS][NAMELENGTH] =
        {"tulip", "marigld", "petunia", "rose", "daisy"};
    char * index[NUMFLOWERS];
    char * t;
    int i, j;
    /* Set up addresses in the array to point to actual strings */
    for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre>
    for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);
    for (i=0; i<NUMFLOWERS; i++) {</pre>
        for (j=i+1; j<NUMFLOWERS; j++) {</pre>
            if (strcmp(index[i], index[j]) > 0) {
                t = index[i];
                index[i] = index[j];
                index[j] = t;
    for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);
```

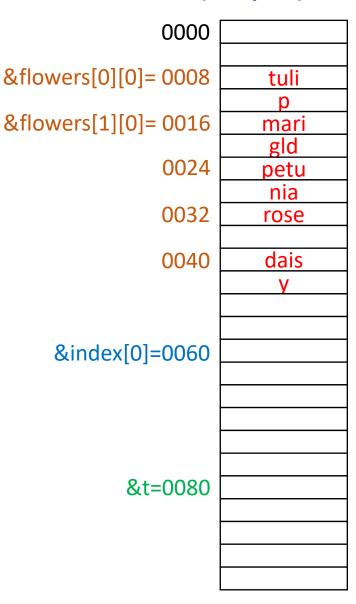
0000	
&flowers[0][0]= 0008	tuli
	р
&flowers[1][0]= 0016	mari
	gld
0024	petu
	nia
0032	rose
0040	dais
	У

Initialization of the string array...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOW __ NAMELENGTH] = {"tulip", "mar, "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up addresses in the array to point to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

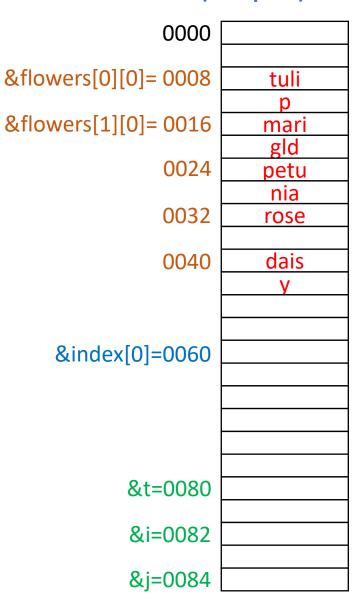


#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[N/ _OWERS][NAMELENGTH] = marigld", "petunia", "rose", "daisy"}; {"tuli char * in NUMFLOWERS]; char * t; int i, j; /* Set up addresses in the array to point to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);



Declaration of temporary pointer for swap...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "mar ld", "petunia", "rose", "daisy"}; char * t; int i, j; /* Set up addresses in the array to point to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);



#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up esses in the array to point to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

tuli
mari
gld petu
nia rose
dais
У
0

Initialization of the address array pointing to the strings ...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up addresses in the array to / ____ to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

0000	
&flowers[0][0]= 0008	tuli
&flowers[1][0]= 0016	p mari
0024	gld petu
0032	nia rose
0040	dais
	У
&index[0]=0060	0008
&t=0080	
&i=0082	0
&j=0084	

Initialization of the address array pointing to the strings ...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up addresses in the array to / ____ to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

0000	
&flowers[0][0]= 0008	tuli
&flowers[1][0]= 0016	p mari
0024	gld petu
0032	nia rose
0040	dais
	У
&index[0]=0060	0008 0016
&t=0080	
&i=0082	1
&j=0084	

Initialization of the address array pointing to the strings ...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j addresses in the array to point to actual strings */ for ; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

0000	
&flowers[0][0]= 0008	tuli
&flowers[1][0]= 0016	p mari
	gld
0024	petu
0032	nia rose
0040	dais
	У
&index[0]=0060	0008
	0016 0024
	0032
	0040
&t=0080	
&i=0082	5
&j=0084	

Array initialization finished...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up ses in the array to point to actual strings */ for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

Abstract Memory Managed by Runtime (Compiler)

0000	
&flowers[0][0]= 0008	tuli
&flowers[1][0]= 0016	p mari
	gld
0024	petu
0032	nia rose
0040	dais
	У
&index[0]=0060	0008 0016
	0024
	0032
	0040
&t=0080	
&i=0082	0
&j=0084	

Now printing the string using the array of addresses...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up addre in the array to point to actual strings */ for (i=0; i LOWERS; i++) index[i] = flowers[i]; for (i=0; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) { for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

0000	
&flowers[0][0]= 0008	tuli
0.61	р
&flowers[1][0]= 0016	mari
0024	gld
0024	petu
0022	nia
0032	rose
0040	dais
	V
&index[0]=0060	0008 0016 0024 0032 0040
&t=0080	
&i=0082	0
&j=0084	

Starting the sorting...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up addresses in the array to point to actual strings */ for (i=0; i<NUMFLOW , i++) index[i] = flowers[i]; for (i=0; i<N/ for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

0000	
&flowers[0][0]= 0008	tuli
&flowers[1][0]= 0016	p mari
0024	gld petu
	nia
0032	rose
0040	dais
	У
&index[0]=0060	0008
	0016
	0032
	0040
&t=0080	
&i=0082	0
&j=0084	1

Second loop...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up addresses in the array to point to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) { for (j=i+1; j<NUMFLOWERS; j if (strcmp(index[i] __dex[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

0000	
&flowers[0][0]= 0008	tuli
	р
&flowers[1][0]= 0016	mari
	gld
0024	petu
	nia
0032	rose
0040	al a ! a
0040	dais
	У
&index[0]=0060	0008
Williacy[0]=0000	0016
	0024
	0032
	0040
&t=0080	8000
• • • • • • • • • • • • • • • • • • • •	
&i=0082	0
0:_0004	1
&j=0084	1

First part of the swap...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up addresses in the array to point to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) s) t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

0000	
&flowers[0][0]= 0008	tuli
	р
&flowers[1][0]= 0016	mari
	gld
0024	petu
	nia
0032	rose
0040	dais
0040	dais
	У
&index[0]=0060	0016
απαελ[0]-0000	0016
	0024
	0032
	0040
&t=0080	0008
_	
&i=0082	0
0: 0004	4
&j=0084	1

Middle part of the swap...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up addresses in the array to point to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[i]) > 0) { t = index[i]; index[i] = index index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

0000	
&flowers[0][0]= 0008	tuli
	р
&flowers[1][0]= 0016	mari
	gld
0024	petu
	nia
0032	rose
0040	daic
0040	dais
	У
&index[0]=0060	0016
απαεκίοι σοσο	0008
	0024
	0032
	0040
&t=0080	0008
0: 0000	
&i=0082	0
Q.i_0001	1
&j=0084	1

Last part of the swap...

#define NAMELENGTH 8 #define NUMFLOWERS 5 void test() { char flowers[NUMFLOWERS][NAMELENGTH] = {"tulip", "marigld", "petunia", "rose", "daisy"}; char * index[NUMFLOWERS]; char * t; int i, j; /* Set up addresses in the array to point to actual strings */ for (i=0; i<NUMFLOWERS; i++) index[i] = flowers[i];</pre> for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]); for (i=0; i<NUMFLOWERS; i++) {</pre> for (j=i+1; j<NUMFLOWERS; j++) {</pre> if (strcmp(index[i], index[j]) > 0) { t = index[i]; index[i] = index[j]; index[j] = t; for (i=0; i<n; i++) printf(" %X: %s\n", index[i], index[i]);

0000	
0000	
&flowers[0][0]= 0008	tuli
	р
&flowers[1][0]= 0016	mari
0004	gld
0024	petu
0022	nia
0032	rose
0040	dais
0040	V
	,
&index[0]=0060	0040
	0016
	0024
	0032
	0008
0 + _0000	0000
&t=0080	8000
&i=0082	0
Q1-000Z	<u> </u>
&j=0084	1

After all iterations...