



EECS 22: Advanced C Programming Assignment 5

Guantao Liu
guantaol@uci.edu

11/18/2016



Outlines

- Double Linked List Traversal
- Command-Line Argument Parsing
- Spotlight Movement

Double Linked List Traversal

SaveMovie() from MovieLab.c:

```

218 count = 0;
219 curr = movie->Frames->First;
220 while (curr != NULL) {
221     for (y = 0; y < curr->YUVImage->Height; y++) {
222         for (x = 0; x < curr->YUVImage->Width; x++) {
223             fputc(GetPixelY(curr->YUVImage, x, y), file);
224         }
225     }
226
227     for (y = 0; y < curr->YUVImage->Height; y += 2) {
228         for (x = 0; x < curr->YUVImage->Width; x += 2) {
229             fputc(GetPixelU(curr->YUVImage, x, y), file);
230         }
231     }
232
233     for (y = 0; y < curr->YUVImage->Height; y += 2) {
234         for (x = 0; x < curr->YUVImage->Width; x += 2) {
235             fputc(GetPixelV(curr->YUVImage, x, y), file);
236         }
237     }
238
239     curr = curr->Next;
240     count++;
241 }

```

Initialization Not reaching the end of the linked list

Operations on the Current Entry

Move to the Next Entry

Double Linked List Traversal

- Variations:
 - What if you need to add an entry at the end?
 - AppendRGBImage
 - AppendYUVImage
 - What if you need to delete some entries?
 - CropImageList: at either end
 - FastImageList: in the middle
 - What if you need to reverse the order?
 - ReverseImageList
- Hints:
 - Use extra pointers (in addition to curr)
 - Use additional variables to keep track of the position

Command-Line Argument Parsing

Main() from MovieLab.c:

```

40 while (x < argc) {
41     /* the input file name */
42     if (strcmp(argv[x], "-i") == 0) {
43         if (x < argc - 1) {
44             finLen = strlen(argv[x + 1]) + strlen(".yuv") + 1;
45             fin = (char *)malloc(sizeof(char) * finLen);
46             if (fin == NULL) {
47                 printf("Error in memory allocation for the input file name!\n");
48                 free(fout);
49                 return 5;
50             }
51             strcpy(fin, argv[x + 1]);
52             strcat(fin, ".yuv");
53         } /*if*/
54     } else {
55         printf("Missing argument for the input file name!\n");
56         free(fin);
57         free(fout);
58         return 5;
59     } /*else*/
60     x += 2;
61     continue;
62 } /*if*/
94
95 x++;
96 } /*elihw*/

```

Copy the input file name from the command line

Why? #1

Error Handling

Why? #2

Why? #3

Why? #4

Command-Line Argument Parsing

- Answers:
 1. The “-i” option always requires a second argument for the input file name after it.
 2. If successful, we have processed two arguments in the command line.
 3. Now start from the beginning of the while loop to check any possible option.
 4. Skip any unrecognized option.

