

School of Computer Science and Software Engineering
Clayton Campus, Monash University
CSE1303 Part A
Summer Semester, 2002

Tutorial 3: Lists, ADTs, and Dynamic Memory

Please attempt all starred questions before your tutorial class.

Exercise 1.*

Write a C declaration for a list of **20** items of type **float**, which can be used in an array implementation of a linked list.

Exercise 2.*

Given

```
struct ItemRec
{
    int number;
    char name[80];
};

typedef struct ItemRec Item;

Item* myItems;
```

Write a C statement so that **myItems** points to a dynamically created array of **10** structures of type **Item**.

Exercise 3.*

Write a C function **char* strdup(const char* s)** which makes a copy of the string **s** and returns a pointer to where the copy is stored, and **NULL** if there is not enough memory.

Exercise 4.*

The program on the next page, which counts characters, words, and lines from the standard input. Rewrite the program so that the only variables are pointers, and make sure any memory allocated during the program is deallocated.

Exercise 5. (Will not be on the exam)

Identify Abstract Data Types in a computer game.

Exercise 6.

Modify the declaration of a stack and the function **initializeStack** given in lectures so that the stack can store any number of items of type **float**, and the maximum number of items it does store is determined when the stack is initialized.

```
/* This program is based on a program in
 * "The C Programming Language",
 * by Brian W. Kerninghan and Dennis M. Richie.
 */

#include <stdio.h>

#define IN 1 /* inside a word */
#define OUT 0 /* outside a word */

/* Count lines, words, and characters in the input */

void main()
{
    int c;
    int state;
    long nl;
    long nw;
    long nc;

    state = OUT;
    nl = nw = nc = 0;

    c = getchar();
    while (c != EOF)
    {
        nc++;
        if (c == '\n')
        {
            nl++;
        }

        if (c == ' ' || c == '\n' || c == '\t')
        {
            state = OUT;
        }
        else if (state == OUT)
        {
            state = IN;
            nw++;
        }

        c = getchar();
    }
    printf("%ld %ld %ld\n", nl, nw, nc);
}
```

Additional Exercises

Kruse et al.

Exercises 4.5: E1 a, b, i, j, n, E2, E4, E5

Review Questions Chapter 4: 5, 8, 20

Exercises 5.2: E1, E2

Deitel & Deitel (2e)

Self Review Exercises: 12.1 a, b, g