Solution: Mid Term Exam Review

This document is compiled to review some main concepts for the coming exam. Exam will include all contents from Chapter 5 to Chapter 8.

Exercise 1 (Functions Review):

For solution see 5.11 Example: A Game of Chance from textbook.

Exercise 2 (Arrays and Pointer Review):

For solution see Case Study: Card Shuffling and Dealing Simulation from textbook, chapter 7.

Exercise 3 (Characters and Strings Review):

(Tokenizing Telephone Numbers) Write a program that inputs a telephone number as a string in the form (555) 555-5555. The program should use function strtok to extract the area code as a token, the first three digits of the phone number as a token and the last four digits of the phone number as a token. The seven digits of the phone number should be concatenated into one string.

The program should convert the area-code string to int and convert the phone-number string to long.

Solution:

```
#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int main()

{

charp[ 20];  /* complete phone number */

charphoneNumber[ 10] = { '\0'}; /* long integer phone number */

char*tokenPtr;  /* store temporary token */

intareaCode;  /* store area code */

longphone;  /* store phone number */

printf( "Enter a phone number in the form ( 555 )" " 555-5555:\n");

gets( p );

/* convert area code token to an integer */

areaCode = atoi( strtok( p, "()") );
```

```
 /* take next token and copy to phoneNumber */

tokenPtr = strtok( NULL, " -" );

strcpy( phoneNumber, tokenPtr );

/* take last token and concatenate to phoneNumber */

tokenPtr = strtok( NULL, "" );

strcat( phoneNumber, tokenPtr );

/* convert phoneNumber to long integer */

phone = atol( phoneNumber );

printf( "\nThe integer area code is %d\n", areaCode );

printf( "The long integer phone number is %ld\n", phone );

return 0; /* indicate successful termination */

} /* end main * Both the area code and the phone number should be printed.
```

Exercise 4 (Characters and Strings Review):

(Removing a Particular Word From a Given Line of Text) Write a program that inputs a line of text and a given word. The program should use string library functions strcmp and strcpy to remove all occurrences of the given word from the input line of text. The program should also count the number of words in the given line of text before and after removing the given word using the strtok function.

Solution:

```
#include<stdio.h>

#include<string.h>

#include<ctype.h>

int main()

{

chartext[ 3][ 80];  /* 3 lines of text */

char*searchPtr;  /* pointer to search character */

charcharacters[ 26] = { 0}; /* totals for each letter */

int count = 0;  /* total for current letter */

int i;  /* loop counter */

int j;  /* loop counter */

printf( "Enter three lines of text:\n");

/* read three lines of text */

for( i = 0; i <= 2; i++ ) {
```

```
gets( &text[ i ][ 0] );
} /* end for */
/* convert letters to lowercase */
for ( i = 0; i <= 2; i++ ) {
/* loop through each character of line */
for( j = 0; text[ i ][ j ] != '\0'; j++ ) {
text[ i ][ j ] = tolower( text[ i ][ j ] );
} /* end for */
} /* end for */
/* loop through alphabet */
for( i = 0; i <= 25; i++ ) {
/* loop through 3 lines of text */
for( j = 0, count = 0; j <= 2; j++ ) {
searchPtr = &text[ j ][ 0];
/* while strchr does not return NULL */
while( searchPtr = strchr( searchPtr, 'a'+ i ) ) {
++count;
searchPtr++;
} /* end while */
} /* end for */
characters[ i ] = count;
} /* end for */
printf( "\nThe total occurrences of each character:\n");
/* display totals for each character */
 for( i = 0; i <= 25; i++ ) {
printf( "%c:%3d\n", 'a'+ i, characters[ i ] );
} /* end for */
return 0; /* indicate successful termination */
} /* end main */
```

Exercise 5 (Characters and Strings Review):

(Printing Dates in Various Formats) Dates are commonly printed in several different for-mats in business correspondence. Two of the more common formats are

07/21/2003 and July 21, 2003

Write a program that reads a date in the first format and prints it in the second format.

Solution:

```
void main(){
char[3] month_form_1;
int year = 2003;
int date = 21;
int month_form_1 = 7;
switch(month){
case 1:
  month_form_1 = "Jan"
  break;
case 2:
  month_form_1 = "Feb"
  break;
  ....
  }
printf("date: %s %d, %d",&month_form_1,&date,&year);
}
}
```