OFFICIAL EXAMINATION
MIDTERM # 2 – November 24th 2003

From: The Code of Academic Integrity

Academic dishonesty is a serious offense which may result in suspension or expulsion from the University. In addition to any other action taken, such as suspension or expulsion, the grade “XF” denoting “failure due to academic dishonesty” will normally be recorded on the transcripts of students found responsible for acts of academic dishonesty.

Students are encouraged to report academic dishonesty. Dial 314-8206 and ask for the Campus Advocate.

EXAMINATION RULES

• Place final answers for each question onto the appropriate underlined blanks. You will not receive full credit even if you have the right answer if it isn’t placed in the right place.
• This examination is CLOSED BOOKS and NO CALCULATORS are allowed.
• Please PRINT legibly. If your work isn’t clear, it may be marked wrong.
• Put your name and recitation number on top of each page.
• Show your work for all questions directly on this exam in order to receive partial credit (where appropriate).
• If you need more space than provided, use the back side of these pages and indicate appropriately.
• There are 7 pages (including this sheet) for this exam, make sure that all are present. Notify instructor immediately if a page is missing.
Question 1 [25 Points]

[4] Fill In The Blanks

1) [2] Write the UNIX command to run the executable a.out that reads data from the input file “input.txt” rather than have the user enter the data via the keyboard.

2) [2] Strings are _______ arrays with the property that they always end with ____________.

[16] Short Answer: Provide 2-3 sentences for the following three questions.

1) [4] Explain the difference between call-by-reference and call-by-value. Illustrate this via an example of a function prototype (foo) that takes two float parameters: one by reference and one by value.

2) [4] Give two differences between a local variable and a global variable. Illustrate this via declaring a local variable ‘a’ and a global variable ‘b’ for a program with just the main function.
3) [3] Why are function prototypes used?

4) [10] Given the following declarations:

```c
int val = 3;
int nums[] = { 1, 2, 3, 4, 5, 6};
char *str = "Hello World";
int *ptr = &nums[1];
```

Provide the data types for the following statements, valid choices are: integer, character, integer memory address, character memory address, and syntax error.

a) nums: ____________________________
b) *(str + 4): ______________________
c) *val: ____________________________
d) nums[4]: _________________________
e) ptr[3]: __________________________
f) &str[1]: _________________________
g) *&nums: _________________________
h) &val: __________________________
i) *ptr: ____________________________
j) ptr + 4: _________________________
Question 2 - Functions [20 Points]: Given the following functions, give the EXACT output for the following segments of code.

```
#include <stdio.h>

int val = 6;
int main ( void ) {
  int a=2, y=3, z=4;
  int result;
  result = foo1( a, y);
  printf("%d %d %d %d\n", result, a, y, val);
  return 0;
}

int foo1( int a, int b ) {
  int val;
  val = a;
  a = b;
  b = val;
  return (2*a+b);
}

int foo2( int *a ) {
  int *p = a;
  (*p)++;
  *a += 4;
  return *a;
}
```

(a) [5]
```
#include <stdio.h>

int main ( void ) {
  int a=2, y=3, z=4;
  int result;
  result = foo1( a, y);
  printf("%d %d %d %d\n", result, a, y, val);
  return 0;
}
```

(b) [5]
```
#include <stdio.h>
#define N 4

int main ( void ) {
  int a=2, y=3, z=4;
  int result;
  result = y + foo2( &z );
  printf("%d %d %d\n", result, z);
  return 0;
}
```
```c
int * foo4( int *ptr, int size ) {
    int tmp = size;
    val = 0;
    while( size-- ) {
        val += *ptr++;
    }
    return (ptr - tmp/2);
}
```

```c
int foo5( int a, int b ) {
    if( !a || !b ) {
        return (a-b);
    }
    return a + b + foo5(a-1, b-1);
}
```

(c) [5]
#include <stdio.h>
#define N 4
int val = 6;

int main ( void ) {
    int nums[ N ] = { 2, 4, 5, 7 };
    int *ptr;
    ptr = foo4( nums, N );
    printf("%d %d\n", val, *ptr);
    return 0;
}

(d) [5]
#include <stdio.h>

int main ( void ) {
    int y=3, z=4;
    result = foo5( y, z );
    printf("%d\n", result );
}
Question 3 - Pointers [20 Points]: Give the output for the following code segments. Each section is independent of all other sections. A “pictorial” description of the array is provided for your ease.

int nums[] = { 1, 3, -1, 4, 8, 2, 7, 5, -2, 6 };
int *ptr1 = nums + 9, *ptr2 = &nums[4], *ptr3 = NULL;
int cnt;

(a) [5]

(*nums+4))++;  
*ptr1 += nums[4];
ptr3 = ptr2++;
*ptr2 += *nums + *ptr3;

printf("%d %d %d\n", *ptr1, *ptr2, *ptr3);

(b) [5]

cnt = 0;
ptr3 = &nums[ ptr1 - ptr2 ];
if( *ptr3 < *ptr2 )
    *ptr2 = *(ptr1-2);
while( *ptr2++ < *ptr1 ) cnt++;
ptr2 = nums + cnt;

printf("%d %d %d\n", *ptr1, *ptr2, *ptr3);

(c) [5]

ptr3 = &ptr2[-4];
*ptr3 = *(nums + *ptr1);
*ptr2 = --ptr1;
ptr1 += 3 * *ptr2;

printf("%d %d %d\n", *ptr1, *ptr2, *ptr3);

(d) [5]

*ptr2 = *ptr1++;
ptr3 = &ptr1;
ptr1 = ptr2 - *ptr2/2;
ptr2 = &nums[ ptr1[ *ptr1 ] ];

printf("%d %d %d\n", *ptr1, *ptr2, *ptr3);
Question 4:  Programming Question – Strings [35 points]

(a) [2] Write the prototype for a function mystrstr that takes in as parameters two strings and returns a character pointer.

(b) [12] The function my strstr locates the first occurrence of the second string parameter in the first string. If found it returns a pointer to the beginning of that string in the first string, else returns a NULL.

Example:  mystrstr( “Hello World”, “World” ); returns a pointer to the seventh character in the string “Hello World”.

Implement the function mystrstr using any string library function besides the strstr string library function. A list of string library functions that you may use are listed at the end of this exam.
(c) [2] Write the prototype for the function stringReplace. It takes in as parameters three strings, and doesn't return anything.

(d) [10] The stringReplace function replaces ALL occurrences of the second string in the first string by the third string. Implement the function stringReplace that uses the help of your string library function `mystrstr`. Assume that the second and third strings are of the same length. You may use any of the string library functions listed at the end of this exam.

Example: `stringReplace( "Hello World, Hello", "lo", "py" );`  
This function call will replace all occurrences of the string "lo" with "py", thus the resulting string will be "Helpy World, Helpy"."
(e) [9] Write a function main that declares three strings, each of which can be a maximum of 20 characters long. It acquires from the user the input string, the query string, and the replacement string. If the query string and the replacement string are of the same length, it should call your function stringReplace and then print the modified string, else it should exit the program.

Example:

Input String:  Hello World Hello
Query String:  lo
Replace With:  py

Output: Helpy World Helpy
String Library Functions:

// Append string SRC to string DST
char *strcat(char *DST, const char *SRC);

// Append LENGTH amount of characters from SRC to string DST
char *strncat(char *DST, const char *SRC, int LENGTH);

// Compare string A to string B, if strings are equal returns a 0, non-zero otherwise
int strcmp(const char *A, const char *B);

// Compare the first LENGTH amount of characters of strings A and string B, if the first LENGTH characters match, returns a 0, non-zero otherwise
int strncmp(const char *A, const char *B, int LENGTH);

// Copy the SRC string into the DST string
char *strcpy(char *DST, const char *SRC);

// Copy the first LENGTH amount of characters from SRC string into the DST string
char *strncpy(char *DST, const char *SRC, int LENGTH);

// Returns the length of the NULL terminated string STR
// Length returned DOES NOT include the NULL character
int strlen(const char *STR);