CSE 220 – Programming in C

Quiz #2

Spring 2016

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Specify if each of the following statements is True or False (4 pts each).

|  |  |
| --- | --- |
| If p and q are two pointers to elements of the same array, subtracting p from q is a valid arithmetic operation |  |
| When a function is called, its parameters are local to the function. |  |
| When an array is passed to a function, copies of the array elements are made for the function to manipulate. |  |
| If p is a pointer to an int, then the expression &p is invalid and results in a compiler error. |  |
| A function can return more than one value at a time |  |

1. Consider the following array. Write statements to do the following: (15 pts)

double a[10] = {0.0, 10.0, 20.0, 30.0, 40.0, 21.0}, \*p, \*q;

|  |  |
| --- | --- |
| Set p to point to the last element of the array |  |
| Set q to point to 2 elements before the element p points to |  |
| Use p to set the second to last element in the array to -5 |  |
| Let q point to the same element as p |  |
| Print the address of p |  |
| Using p, print the number of elements separating \*p from the first element of a |  |
| Write an if statement that prints “YES” if the value that p points to is smaller than the value that q points to and “NO” otherwise | |

1. Write a recursive function that takes an array of char and returns 1 if all characters it contains are letters from the alphabet (a to z or A to Z), and 0 otherwise. (15 pts):
2. What is the output of the following c program (21 pts).

#include <stdio.h>

int a;

void addOne(int a) {

a++;

printf(“W. a = %d\n”, a);

}

void removeOne(int a) {

int b = a + 1;

printf(“R. b = %d\n”, b);

}

void swap(int \*a, int \*b) {

int temp = \*a;

\*a = \*b;

\*b = temp;

}

int main() {

a = 5;

int b = 20;

if (b > 15) {

int a = 53;

removeOne(b);

addOne(a);

printf(“X. a = %d\n”, a);

}

printf(“Y. a = %d, b = %d\n”, a, b);

swap(&a, &b);

printf(“Z. a = %d, b = %d\n”, a, b);

return 0;

}

1. The function lowerTriangular takes as input a two dimensional array and its size and returns 1 if the array represents a lower triangular matrix and 0 otherwise. A two dimensional array is lower triangular if the number of columns is equal to the number of rows and all elements above the diagonal are zero. Finish the following program (29 pts)

/\* include any libraries needed \*/ (2)

/\* declare the function lowerTriangular (prototype only) \*/ (3)

int main(){

/\* declare a 25x25 2-dimensional array \*/ (3)

/\* Initialize the arrays to random numbers \*/ (5)

/\* Call lowerTriangular on the array and print the result \*/(5)

}

/\* define the function lowerTriangular \*/ (11)