Name: Eland Anthony

CS403/503 Programming Languages
Spring 2021
Assignment #4

Note: if it asks you to write a program, make sure the program compiles and executes.

1. Write a program in C that uses the `qsort` function to sort an array of real numbers (`double`). Paste below only the comparison function passed to `qsort` as a parameter.

Answer:

```
int comparator(const void* ptr, const void* ptr1) {
return ((*(double*)ptr) - (*(double*)ptr1));
}
```

2. Problem 5 of Chapter 9 on Page 413:

Consider the following program written in C syntax:

```
void swap(int a, int b) {
int temp;
temp = a;
a = b;
b = temp;
}
void main() {
int value = 2, list[5] = {1, 3, 5, 7, 9};
swap(value, list[0]);
swap(list[0], list[1]);
swap(value, list[value]);
}
```

For each of the following parameter-passing methods, what are all of the values of the variables value and list after each of the three calls toswap?

1. Passed by value
2. Passed by reference
3. Passed by value-result

Answer:

Name: Eland Anthony

Number	Value	List Contents
1	2	1, 3, 5, 7, 9
2	2	1, 3, 5, 7, 9
3	2	1, 3, 5, 7, 9

Number	Value	List Contents
1	1	2, 3, 5, 7, 9
2	1	3, 2, 5, 7, 9
3	2	3, 1, 5, 7, 9

Number	Value	List Contents
1	1	2, 3, 5, 7, 9
2	1	3, 2, 5, 7, 9
3	2	3, 1, 5, 7, 9

3. Programming Exercise 8 of Chapter 9 on Page 415:

Write a generic C++ function that takes an array of generic elements and a scalar of the same type as the array elements. The type of the array elements and the scalar is the generic parameter. The function must search the given array for the given scalar and return the subscript of the scalar in the array. If the scalar is not in the array, the function must return –1. Test the function for int and float types.

Answer:

Name: Eland Anthony

A4P3.h

```
template <class Type>
int generic_search(Type list[3], Type num) {
    for (int i = 0; i <= 2; i++) {
        if (num == list[i])
            return i;
        }
    return -1;
}
```

A4P3.cpp

```
#include <stdio.h>
#include <iostream>
#include "A4P3.h"

using namespace std;
```

```
int main() {
    int intarr[3];
    float floatarr[3];
    int a;
    float b;
    cout << "Enter three integer elements and separate by
space\n";
    for (int i = 0; i <= 2; i++) {
        cin >> intarr[i];
    }
    cout << "Enter the int search element\n";
    cin >> a;
    int output = generic_search(intarr, a);
    if (output == -1)
        cout << "Not present\n";
    else
        cout << output << " Present\n";
```

Name: Eland Anthony

```
cout << "Enter three float elements separated by space\n";
for (int j = 0; j <= 2; j++) {
     cin >> floatarr[j];
     }
cout << "Enter the float search element\n";
cin >> b;
int output1 = generic_search(floatarr, b);
if (output1 == -1)
     cout << "Not present\n";
else cout << output1 << " Present\n";
return 0;
}
```

4. Problem 2 of Chapter 10 on Page 443:

Show the stack with all activation record instances, including static and dynamic chains, when execution reaches position 1 in the following skeletal program. Assume bigsub is at level 1.

```
function bigsub() {
var mysum;
function a() {
var x;
function b(sum) {
var y, z;
...
c(z);
...
} // end of b
...
b(x);
...} // end of a
function c(plums) {
... <----------------------------1
} // end of c
var l;
...
a();
...
} // end of bigsub
```

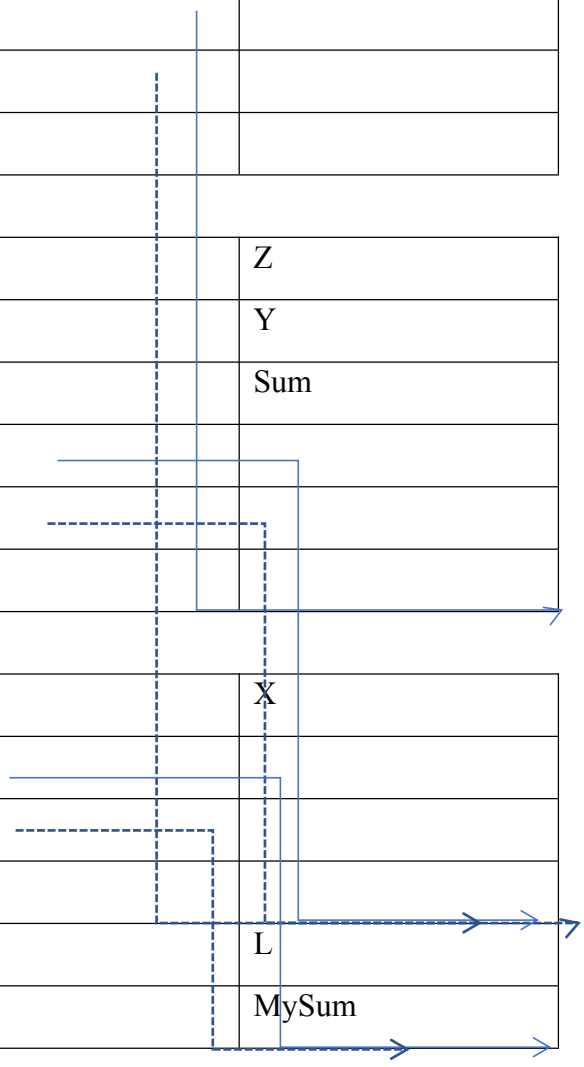
Name: Eland Anthony

Answer: Top

ARI for C	Parameter		Plums
	Dynamic link		
	Static link		
	Return (to B)		

ARI for B	Local		Z
	Local		Y
	Parameter		Sum
	Dynamic link		
	Static link		
	Return (to A)		

ARI for A	Local		X
	Dynamic link		
	Static link		
	Return (to Bigsub)		
ARI Bigsub	Local		L
	Local		MySum



ARI = Activation Record Instance