Object Oriented Software Development

Pointers

Synopsis

This laboratory session is intended to enable you understand ***pointers*** and related concepts. If you have not done yet, watch the lecture before you attempt this seminar (or do both in parallel).

Answers are provided at the end of this file.

Exercises

1. Write a program to demonstrate the use of dereferencing operator.

2. Write a program to sum the elements of a dynamic array using pointers.

3. Write a program that reads a group of numbers from the user and places them in an array of type float. Once numbers are stored in the array, the program should average them and print the result. Use pointer notation wherever possible.

4. Start with the String class as given below. Add a member function called upit() that converts the string to all uppercase. You can use the toupper() library function, which takes a single character as an argument and returns a character that has been converted (if necessary) to uppercase. This function uses the CCTYPE header file. Write some code in main() to test upit().

```
//Using new to get memory for strings
#include <iostream>
#include <cstring>   //for strcpy(), etc
using namespace std;

class String //user-defined string type
{
        private:
        char* str; //pointer to string
        public:
        String(char* s) //constructor, one arg
        {
                int length = strlen(s); //length of string argument
                str = new char[length+1]; //get memory
                strcpy(str, s); //copy argument to it
        }
        ~String() //destructor
        {
                cout << "Deleting str.\n";
                delete[] str; //release memory
        }
        void display() //display the String
        {
                cout << str << endl;
        }
};

int main()
{
        //uses 1-arg constructor
        String s1 = "Who knows nothing doubts nothing.";

        cout << "s1="; //display string
        s1.display();
        return 0;
}
```

5. Amend the program on slide no 30 to add a function 'multiply' similar to the functions 'addition' and 'subtraction' already there. Add code in main function to call this newly added function.

Answers

1. Refer to lecture slide 9

Object Oriented Software Development

Pointers

2. See lecture slides 11, 16, 23 and 25

3.

```
//Finds average of numbers typed by user

#include <iostream>
using namespace std;
int main()
{
	float flarr[100]; //array for numbers
	char ch; //user decision
	int num = 0; //counts numbers input
	do
	{
		cout << "Enter number: "; //get numbers from user
		cin >> *(flarr+num++); //until user answers 'n'
		cout << " Enter another (y/n)? ";
		cin >> ch;
	}
	while(ch != 'n');

	float total = 0.0; //total starts at 0
	for(int k=0; k<num; k++) //add numbers to total
	total += *(flarr+k);

	float average = total / num; //find and display average
	cout << "Average is " << average << endl;
	return 0;
}
```

4.

```
#include <iostream>
#include <cstring> //for strcpy(), etc
#include <cctype> //for toupper()
using namespace std;

class String //user-defined string type
{
	private:
	char* str; //pointer to string
	public:
	String(char* s) //constructor, one arg
	{
		int length = strlen(s); //length of string argument
		str = new char[length+1]; //get memory
		strcpy(str, s); //copy argument to it
	}
	~String() //destructor
	{
		delete str;
	}
	void display() //display the String
	{
		cout << str;
	}
	void upit(); //uppercase the String
};


void String::upit() //uppercase each character
```

Pointers

```
{
        char* ptrch = str; //pointer to this string
        while( *ptrch ) //until null,
        {
                *ptrch = toupper(*ptrch); //uppercase each character
                ptrch++; //move to next character
        }
}

int main()
{
        String s1 = “He who laughs last laughs best.”;
        cout << “\ns1=”; //display string
        s1.display();
        s1.upit(); //uppercase string
        cout << “\ns1=”; //display string
        s1.display();
        cout << endl;
        return 0;
}
```

5.

```
#include <iostream>
#include <cstring> //for strcpy(), etc
#include <cctype> //for toupper()
using namespace std;

int multiply(int a, int b){
return a*b;
}


int main(){


result=operation(2,3, multiply);




}
```