**Karem Abed Naser Ramsis**
**Computer engineering - Object oriented programming[eng] R032**

**Problem 1:**

```
#include <iostream>
using namespace std;

void print_smallest_and_largest(int a[10]) // function to get cmallest and largest
{
   int smallest, smallest_index, largest, largest_index; // creating local function
variables for all 4 wanted values

   int i = 0; // initialising i
   smallest_index = i; // using i for the index
   largest_index = i; // using i for the index
   smallest = a[i];// using a[i] for the value of the index
   largest = a[i];// using a[i] for the value of the index

   for (i; i < 10; i++) {
// for loop to cycle through the array to get the larges and smalles and their index in
the array

      if (a[i] < smallest) {
         smallest = a[i];
         smallest_index = i;
      }
      if (a[i] > largest) {
         largest = a[i];
         largest_index = i;
      }
}
// outputing the results of the function
   cout << "largest int is " << largest << " at index " << largest_index << endl;
   cout << "smallest int is " << smallest << " at index " << smallest_index << endl;

}

void sort_desc(int a[10])  //sort in descending
{
   int temp; // creating a temp variable

  // creeating a for loop that uses three integers: i,j and temp.
   for (int i = 0; i < 10; i++) // this for loop will increment the internal for loop and
execute it 10 times
```

```
    {
                // a for loop is nested that will use a temporary variable to hold a value while it checks the value and sorts them
        for (int j = i + 1; j < 10; j++)
        {
            if (a[i] < a[j])
            {
                temp = a[i];
                a[i] = a[j];
                a[j] = temp;
            }
        }
    }
}

int main()
{
    int a[10]; // creating a 10 element array of integers

    cout << "Enter 10 Element into the array" << endl; // requesting the user to enter the 10 elements

    for (int i = 0; i < 10; i++) // for loop to add the 10 elements using cin
        cin >> a[i];

    int* p; //pointer used to pass the array to the function
    p = a;

    print_smallest_and_largest(p); // executing the fuction that gets the largest and smallest

    cout << "Here are the sorted array elements in descending order" << endl;
    sort_desc(p); // execute the functions that sorts in descending order
    for (auto x : a) // output the descending order function
        cout << x << endl;

        return 0;
}
```

**Problem 2:**

```
#include <iostream>
#include <string>
using namespace std;
struct professor
// creating the structure,following attributes: First Name, Last Name, Height and age.
{
	string first_name;
	string last_name;
	float height;
	int age;
};
void print (professor a[10])
//function, which prints out the first name and last name of the professor, whos sum of age and height is the biggest.
{
	int i = 0, biggest_index=i;// initializing i and assigning it to biggest index
	float biggest = a[i].height + a[i].age; // defining bigggest as the sum of age and height

	for (i = 1; i < 10; i++) { // for loop to get the biggest
		if ((a[i].height + a[i].age) > biggest) {
			biggest = a[i].height + a[i].age;
			biggest_index = i;
		}
	}
	cout << a[biggest_index].first_name << " " << a[biggest_index].last_name << endl; // printing the biggest index
}

int main()
{
	// the 10 element array
	professor a[10];

	string fname[10] = { "Kevin","Mark","Mary","Katy","Ayla", "Charles","Toby","Lucy", "Shane", "Tom" };
	string lname[10] = { "Duncan","Hart","Smith","Moore","Larson", "Cooper","Simpson","Holt", "White", "Thomas" };
	int age[10] = {37,45,64,45,34,46,35,75,44,56};
	float height[10] = {1.55,1.75,1.45,1.67,1.53,1.78,1.99,1.84,1.76,1.44};// height in meters.centemeters
	//filling the array with random data using a for loop to test
	for (int i = 0; i < 10; i++) {
```

```
            a[i].first_name = fname[i];
            a[i].last_name = lname[i];
            a[i].age = age[i];
            a[i].height = height[i];
        }

        print(a); // calling the function

        return 0;
}
```

**Problem 3:**

//the array should be passed to the function, which will sort the objects in ascending order of the age private variable

```
#include <iostream>
#include <string>
#include <algorithm>
using namespace std;

class p3
{
private:
    string first_name;
    string last_name;
    int age;

public:
    //constructors
    p3() //default constructor
    {
        first_name = "Fname";
        last_name = "Lname";
        age = 21;
    }
   p3(int age_a) //constructor with age only
    {
        first_name = "Fname";
        last_name = "Lname";
        age = age_a;
    }

    p3(string Fname, string Lname) //constructor with first and last name only
    {
        first_name = Fname;
        last_name = Lname;
        age = 21;
    }

    p3(string Fname, int age_a) //constructor with one name only (chose first name)
    {
        first_name = Fname;
        last_name = "Lname";
        age = age_a;
```

```
}

p3(string Fname, string Lname, int age_a) // full constructor
{
   first_name = Fname;
   last_name = Lname;
   age = age_a;
}

// setters
void set_firstName(string f_name)
{
   first_name = f_name;
}
void set_lastName(string l_name)
{
   last_name = l_name;
}
void set_Age(int a_age)
{
   age = a_age;
}
void set_first_Last_names(string Fname, string Lname)
{
   first_name = Fname;
   last_name = Lname;
}
void set_all(string Fname, string Lname, int age_a)
{
   first_name = Fname;
   last_name = Lname;
   age = age_a;
}
//getters
string get_firstName()
{
   return first_name;
}
string get_lastName()
{
   return last_name;
}
int get_Age()
{
   return age;
```

```
    }

    //needed for sorting the aray by age in ascending order
    bool operator<(p3 C) {
        return (age < C.age);
    }

};

int main()
{
    p3 obj[10] = { {"Mary","Lucas",39},{"Joe","Frank",21} ,{"Luis","Clark",55} ,
{"Jessie","Marcus",32} ,{"Bob","Micheal",34} };
    // Initialization of the objects, whos index is less or equal to 4 done using the
constructor

    //the elements of an array whos index is greater than 4,initialized using the set()
function.
    for (int i = 5; i < 9; i++) {
        obj[i].set_firstName("Jon");
        obj[i].set_lastName("Doe");
        obj[i].set_Age(25);
    }
    obj[9].set_all("Hal", "Mich",55);//setting the class using a constructor

    sort(obj, obj + 10); // sorting the array
    // printing the sorted array
    for (auto o : obj) {
        cout << o.get_firstName() << "  " << o.get_lastName() << "  " << o.get_Age()
<< endl;
    }

    return 0;
}
```