



Bahria University, Islamabad

Department of Software Engineering

Computer Programming

(Fall-2023)

Teacher: Dr. Raja Suleman

---

Student : **Atqa asma**

Enrollment : **01-131232-016**

---

Assignment: 1

Date: 29<sup>th</sup> Oct, 2023

---

## **Assignment # 1- Problem Solving**

### **ALGORITHMS**

#### **Question 1: Finding the Shortest Path**

**Step 1:** Start.

**Step 2:** Give names to all points/places.

**Step 3:** Firstly, we just know that the shortest distance of point 1 from any previous point is 0 as it has no previous point and set the shortest distance of all other points an infinity.

**Step 4:** These distance values can be changed overtime.

**Step 5:** Starting from point 1.

**Step 6:** Calculate the distance of point 1 to its unvisited neighbors.

**Step 7:** To find shortest distance from its unvisited points we will update its shortest distance and store point 1 as previous point.

**Step 8:** Now marks point 1 as visited.

**Step 9:** Then, pick a point (point 2) that have shortest distance from previous point.

**Step 10:** Calculate the distance of all unvisited neighbors of point 2 by adding shortest distance of point 2 in it.

**Step 11:** Now, marks the neighbors of point 2 as visited points.

**Step 12:** Compare the calculated distance with one that is previously stored as 'shortest distance' . If the new value is less than previous value than replace it with new one.

**Step 13:** Repeat from step 6 to 12 for other points.

**Step 14:** The points that are once visited are marked 'visited' and we can't go back to them else go to unvisited points.

**Step 15:** For finding shortest path to go from one point to another, we need to add the distances of single points and go toward shortest the path that is shortest one.

**Step 16:** Stop.

#### **Question 2: Sorting a List of Numbers**

**Step 1:** Start.

**Step 2:** Declare a variable i and n.

**Step 3:** Prompt the user to enter how many numbers he want in series(i.e. n).

**Step 4:** Take the series of number from user.

**Step 5:** Use a for loop to iterate from i=0 to i less than n.

**Step 6:** Apply a condition (e.g. for ascending order) that if first element will be greater than second one then do following task.

**Step 7:** Declare third variable.

**Step 8:** Assign it (Third variable) the value of first one.

**Step 9:** Then, assign second value to first one.

**Step 10:** Now, put the value of third variable in second one.

**Step 11:** Repeat steps 8 ,9 and 10 until the condition in step 6 gets false and loop terminates (i.e., when  $i > n$ ).

**Step 12:** Print the series of number that are sorted now.

**Step 13:** Stop.

### **Question 3: Calculating Fibonacci Numbers**

**Step 1:** Start.

**Step 2:** Declare first variable and assign it value 0, second variable and assign it value 1.

**Step 3:** Declare variables i and n.

**Step 4:** Print first and second variables.

**Step 5:** Prompt the user to enter value that how much numbers he wants in series (i.e., value of n).

**Step 6:** Use a for loop to iterate from  $i=1$  to  $i$  less than or equal to  $n-2$ .

**Step 7:** Add first and second variable and store the result in another variable named next.

**Step 8:** Print the variable 'next'.

**Step 9:** Assign the value of second to first variable.

**Step 10:** Then, assign the value of 'next' variable to second.

**Step 11:** Repeat steps 7, 8, 9 and 10 until the loop terminates (i.e., when  $i$  become greater than  $n-2$ ).

**Step 12:** Stop.

### **Question 4: Inventory Management**

**Step 1:** Start.

**Step 2:** Make a database for store's inventory management system.

**Step 3:** For Adding items to the inventory ask the user for name of the item and the quantity of that particular item user want to add.

**Step 4:** Now, check that the item exists before in the list or not. If yes, then update its quantity otherwise add the item along with its quantity in the list.

**Step 5:** For Removing item from the inventory, ask the user for name of the item and the quantity of that particular item user want to remove.

**Step 6:** Now, check that the item exists before in the list or not. If yes, then remove particular quantity of that item or whole. If no, then display a message that 'This Item does not exists'.

**Step 7:** For updating the quantity of the existing item, ask the user for the item name and new quantity of that particular item. Replace new quantity with the old one.

**Step 8:** For generating reports of the items and quantities, we need to use formula according to the type of report we want to generate.

**Step 9:** Stop.

**GitHub link:** <https://github.com/Atqaasma/CP-A-1-Algorithms>