# COMP100-002

# Assignment 1 (Part B)

# Completed By: Chowdhury Atika Parvin (Question 1&2) Joyeeta Nandi (Question 3&4) Qian Long (George) Zhou (Question 5&6) Thomas Cruz (Question 7&8) Jeongmin Kim (Question 9&10)

# 

# **Assignment 1 (Part B) – Due in week 3 at the beginning of your first COMP100 class.**

### **Like your first assignment, you will work in groups of four students. Each group will submit one printout and all members of the group will receive the same mark. Your printout should have a cover page indicating the course code → COMP100-061, the assignment title → Assignment 1 (Part B) and a list of the members of the group.**

**For each of the problems below, identify the output(s) and the inputs. Generate a detailed algorithm (you must prompt for and accept inputs and then display the output on the console) and create a two-column IPO chart with the required algorithm. Your C# statements must align with the matching IPO chart information. Your identifiers must be description and the type must be appropriate to the data that you will be working with. You should desk check your algorithm**

**Checklist:**

* **Did you copy the problem specification to your answer page?**

Penalty – 10 %

Penalty – 5 %

Penalty – 10 %

Penalty – 100 %

Penalty – 5 %

* **Do you have a sample calculation with actual number?**
* **Do you have 2 columns?**
* **Are you using identifiers without spaces?**
* **Are you using a fixed-face font?**
* **Does the IPO statement align with the appropriate c# statement? (You may add empty lines, reduce the size of the fonts and/or adjust the width of the columns)**

Penalty – 20 %

1. After surveying a number of new-home electrical installations, Kelly Builder’s Inc. has worked out what the length of wire a typical house would require. Write a program that will prompt the user for the average length required for a home and the number of houses to wire. The program will then calculate and display the total length of wire required for the specified number of houses.  
     
   Output: totalLength  
   Input: lengthPerHouse, numberOfHouses  
   Sample Calculation:  
   lengthPerHouse\*numberOfHouses = totalLength  
   4\*20 = 80

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| lengthPerHouse numberOfHouses | Algorithm:   1. Prompt and accept lengthPerHouse, numberOfHouses 2. Calculate: totalLength = lenthPerHouse\*numberOfHouses 3. Display totalLength | | totalLength |
| **IPO Chart Info** | | **C# Code** | |
| Input: lengthPerHouse  numberOFHouses  Processing:  Output:  TotalLength  Algorithm:   1. Prompt for lengthPerHouse 2. Accept lengthPerHouse 3. Prompt for numbeOfHouses 4. Accept numberOfHouses 5. Calculate totalLength = lengthPerHouse\*numberOfHouses 6. Display totalLength | | //Input  int length;  int houses;  //Processing  //Output  int total;  //Algorithm  Console.Write("Enter Length: ");  length = Convert.ToInt32(Console.ReadLine());  Console.Write("Enter number of houses: ");  houses = Convert.ToInt32(Console.ReadLine());  total = length \* houses;  Console.WriteLine($"The total length required is {total}"); | |

1. Loreto Farms wants a program to estimate the cost of fertilizing their fields for the coming year. The program will prompt the user for the rate of fertilizer application (how many kg per hectares), the unit price of fertilizer (dollars per kg) and the size (hectares)of land to be cultivated and then calculate and display the cost

Output: cost  
Input: rFert, price, size  
Sample Calculation:  
cost=rFert\*price\*size  
2\*20\*50 = 2000

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| rFert  price  size | Algorithm:   1. Prompt and accept rFert, price, size 2. Calculate: cost = rFert\*price\*size 3. Display cost | | cost |
| **IPO Chart Info** | | **C# Code** | |
| Input: rFert  price  size  Processing:  Output:  cost  Algorithm:   1. Prompt for rFert 2. Accept rFert 3. Prompt for price 4. Accept price 5. Prompt for size 6. Accept size 7. Calculate cost = rFert\*price\*size 8. Display cost | | //Input  int rFert;  int price;  int size;  //Processing  //Output  int cost;  //Algorithm  Console.Write("Enter rate of fertilizer application: ");  rFert = Convert.ToInt32(Console.ReadLine());  Console.Write("Enter unit price for fertilizer: ");  price = Convert.ToInt32(Console.ReadLine());  Console.Write("Enter size of the land: ");  size = Convert.ToInt32(Console.ReadLine());  cost = rFert\*price\*size;  Console.WriteLine($"The total length required is {cost}"); | |

1. Montgomery Entertainment Corp. is looking into the profitability of hosting MMA XXII at the Rogers Center. Write a program that will calculate and display the profit that will result from hosting such as event. The program will calculate the income (you must prompt for each of the following) from the sale of tickets, broadcasting rights and advertising and subtract the cost of renting the Center

Output: profit  
Input: sTicket, bRight, ad, rCost  
Sample Calculation:  
profit=sTicket+bRight+ad-rCost  
20000+5000+5000-10000 = 20000

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| sTicket  bRight  ad  rCost | Algorithm:   1. Prompt and accept sTicket,bRight,ad,rCost 2. Calculate: profit=sTicket+bRight+ad-rCost 3. Display profit | | profit |
| **IPO Chart Info** | | **C# Code** | |
| Input: sTicket  bRight  ad  rCost  Processing:  Output:  profit  Algorithm:   1. Prompt for sTicket 2. Accept sTicket 3. Prompt for bRight 4. Accept bRight 5. Prompt for ad 6. Accept ad 7. Prompt for rCost 8. Accept rCost 9. Calculate profit = sTicket+bright+ad-rCost 10. Display profit | | //Input  int sTicket;  int bRight;  int ad;  int rCost;  //Processing  //Output  int profit;  //Algorithm  Console.Write("Amount of ticket sold: ");  sTicket = Convert.ToInt32(Console.ReadLine());  Console.Write("Broadcasting rights profit: ");  bRight = Convert.ToInt32(Console.ReadLine());  Console.Write("Advertising sales: ");  ad = Convert.ToInt32(Console.ReadLine());  Console.Write("The cost to rent is: ");  rCost = Convert.ToInt32(Console.ReadLine());  profit = sTicket+bRight+ad-rCost;  Console.WriteLine($"The gained from this event is {profit:C}"); | |

1. Narendra Airlines would like a program that will calculate and display the cost of flying an aircraft between various locations. Write a program that will compute the cost which is based on the fuel efficiency (amount of fuel used for each km travelled) of the aircraft, the unit price of fuel and the length of the journey

Output: cost  
Input: fuelEffi, fuelPric, dist  
Sample Calculation:  
cost=fuelEffi\*fuelPric\*dist  
50\*20\*1000 = 1000000

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| fuelEffi  fuelPric  dist | Algorithm:   1. Prompt and accept fuelEffi,fuelPric,dist 2. Calculate: cost=fuelEffi\*fuelPric\*dist 3. Display cost | | Cost |
| **IPO Chart Info** | | **C# Code** | |
| Input: fuelEffi  fuelPric  dist  Processing:  Output:  cost  Algorithm:   1. Prompt for fuelEffi 2. Accept fuelEffi 3. Prompt for fuelPric 4. Accept fuelPric 5. Prompt for dist 6. Accept dist 7. Calculate cost=fuelEffi\*fuelPric\*dist 8. Display cost | | //Input  int fuelEffi;  int fuelPric;  int dist;  //Processing  //Output  int cost;  //Algorithm  Console.Write("The efficiency of this fuel is: ");  fuelEffi = Convert.ToInt32(Console.ReadLine());  Console.Write("The price for the fuel is: ");  fuelPric = Convert.ToInt32(Console.ReadLine());  Console.Write("Travelling distance: ");  dist = Convert.ToInt32(Console.ReadLine());  cost=fuelEffi\*fuelPric\*dist;  Console.WriteLine($"The total cost for this trip is {cost:C}"); | |

1. Othello’s Grocery is small business located in the West Hill area. They would like a program that will prompt the user for the weight and price of the produce, the price and capacity of plastic bag and then calculate and display the total cost of the sale. You may assume that all customers will require bags.

Output: totalCost  
Input: pWeight, pPrice, bPrice, bCap  
Sample Calculation:  
totalCost = pWeight/pCap\*bPrice + pPrice\*pWeight   
totalCost = 20lb/2lb\*$1+$3\*20lb = $10+$60 = $70

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| pWeight  pPrice  bPrice  bCap | Algorithm:   1. Prompt and accept pWeight,pPrice,bPrice,bCap 2. Calculate: totalCost=pWeight/bCap\*bPrice + pPrice+pWeight 3. Display totalCost | | totalCost |
| **IPO Chart Info** | | **C# Code** | |
| Input: pWeight  pPrice  bPrice  bCap  Processing:  Output:  totalCost  Algorithm:   1. Prompt for pWeight 2. Accept pWeight 3. Prompt for pPrice 4. Accept pPrice 5. Prompt for bPrice 6. Accept bPrice 7. Prompt for bCap 8. Accept bCap 9. Calculate totalCost=pWeight/bCap\*bPrice + pPrice+pWeight 10. Display totalCost | | //Input  double pWeight;  double pPrice;  double bPrice;  double bCap;  //Processing  //Output  double totalCost;  //Algorithm  Console.Write("What’s the total weight of the produce: ");  pWeight = Convert.ToDouble(Console.ReadLine());  Console.Write("The price for the produce is: ");  pPrice = Convert.ToDouble(Console.ReadLine());  Console.Write("The price for the bag is: ");  bPrice = Convert.ToDouble(Console.ReadLine());  Console.Write("The capacity for the bag is: ");  bCap = Convert.ToDouble(Console.ReadLine());  totalCost=pWeight/bCap\*bPrice + pPrice+pWeight;  Console.WriteLine($"The total cost for this trip is {totalCost:C}"); | |

1. Parker would like to get each of the females in his family the same gift for the holidays. He has an amount of money which he is willing to use up completely. Write a program that will prompt him for the price of the item and the amount of money that he has and then calculate and display the most items that he can buy and the left-over money. (There are 2 outputs for this question)

Output: amount, remainder  
Input: itemP, budget  
Sample Calculation:

amount = budget/itemP

amount = 5270/100 = 52

remainder = budget – itemP\*amount

remainder = 5270-100\*52 = 70

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| itemP  budget | Algorithm:   1. Prompt and accept itemP,budget 2. Calculate: amount = budget/itemP 3. Calculate: remainder = budget – itemP\*amount 4. Display amount and remainder | | amount  remainder |
| **IPO Chart Info** | | **C# Code** | |
| Input: itemP  budget  Processing:  Output:  amount  remainder  Algorithm:   1. Prompt for itemP 2. Accept itemP 3. Prompt for budget 4. Accept budget 5. Calculate: amount = budget/itemP 6. Calculate: remainder = budget – itemP\*amount 7. Display amount and remainder | | //Input  int itemP;  int budget;  //Processing  //Output  int amount;  int remainder;  //Algorithm  Console.Write("What’s the price of the item: ");  itemP = Convert.ToInt32(Console.ReadLine());  Console.Write("What’s your budget: ");  budget = Convert.ToInt32(Console.ReadLine());  amount = budget/itemP;  remainder = budget-itemP\*amount;  Console.WriteLine($"The amount of items you can buy is {amount}, the remainder left is {remainder:C}"); | |

1. Quincy “The Plumber” bills customer for pipe installation based on the length as well as the number of joins of the job. Build a program that will prompt for length rate, join rate, the length, the number of join and calculate and display the total cost. [cost = (length \* lengthRate) + (numberOfJoins \* joinRate)]

Output: cost  
Input: lPipe,nJoin,cPipe,cJoin  
Sample Calculation:

cost = lPipe\*cPipe + nJoin\*cJoin

cost = 10\*2 + 3\*5 = 20+15=35

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| lPipe  nJoin  cPipe  cJoin | Algorithm:   1. Prompt and accept lPipe,nJoin,cPipe,cJoin 2. Calculate: cost = lPipe\*cPipe + nJoin\*cJoint 3. Display cost | | cost |
| **IPO Chart Info** | | **C# Code** | |
| Input: lPipe  nJoin  cPipe  cJoin  Processing:  Output:  cost  Algorithm:   1. Prompt for lPipe 2. Accept lPipe 3. Prompt for nJoin 4. Accept nJoin 5. Prompt for cPipe 6. Accept cPipe 7. Prompt for cJoin 8. Accept cJoin 9. Calculate: cost = lPipe\*cPipe + nJoin\*cJoin 10. Display cost | | //Input  int lPipe;  int nJoin;  int cPipe;  int cJoin;  //Processing  //Output  int cost;  //Algorithm  Console.Write("What’s the length of the pipe: ");  lPipe = Convert.ToInt32(Console.ReadLine());  Console.Write("How many joins are there: ");  nJoin = Convert.ToInt32(Console.ReadLine());  Console.Write("What’s the cost of the pipe: ");  cPipe = Convert.ToInt32(Console.ReadLine());  Console.Write("What’s the cost of the join: ");  cJoin = Convert.ToInt32(Console.ReadLine());  cost = lPipe\*cPipe + nJoin\*cJoin;  Console.WriteLine($"The total cost is {cost:C}"); | |

1. Last year at the CNE, the Robert Family brought ride tickets. If the Polar Express and the Ferris wheel cost 5 and 3 tickets respectively. Write a program that prompts the user for the amount of ticket bought, the number of times each of the two rides were taken and then calculate how many tickets will remain at the end of the day

Output: remain  
Input: amount, peTime, fwTime  
Sample Calculation:

Remain = amount – 5\*peTime – 3\*fwTime  
Remain = 50 – 5\*3-3\*2 = 50-15-6 = 29

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| amount  peTime  fwTime | Algorithm:   1. Prompt and accept amount,peTime,fwTime 2. Calculate: remain = amount-5\*peTime – 3\*fwTime 3. Display remain | | remain |
| **IPO Chart Info** | | **C# Code** | |
| Input: amount  peTime  fwTime  Processing:  Output:  remain  Algorithm:   1. Prompt for amount 2. Accept amount 3. Prompt for peTime 4. Accept peTime 5. Prompt for fwTime 6. Accept fwTime 7. Calculate: remain = amount-5\*peTime – 3\*fwTime 8. Display remain | | //Input  int amount;  int peTime;  int fwTime;  //Processing  //Output  int remain;  //Algorithm  Console.Write("What’s the total amount of tickets bought: ");  amount = Convert.ToInt32(Console.ReadLine());  Console.Write("How many time has Polar Express been ridden: ");  peTime = Convert.ToInt32(Console.ReadLine()); Console.Write("How many time has Ferris Wheel been ridden: ");  fwTime = Convert.ToInt32(Console.ReadLine());  remain = amount-5\*peTime-3\*fwTime;  Console.WriteLine($"The remaining amount of ticket is {remain}"); | |

1. Stephanie’s Confectionary wants a program that will prompt the user for the price of the candy and the amount of money she would like to spend and then calculates and display the amount of candies that can be purchased as well as the amount of money remaining after purchase.

Output: amount, remain  
Input: cost, budget  
Sample Calculation:

amount = budget/cost = 110/4 = 27  
remain = budget-cost\*amount = 110-4\*27 =2

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| cost  budget | Algorithm:   1. Prompt and accept cost, budget 2. Calculate: amount = budget/cost 3. Calculate: remain = budget – cost\*amount 4. Display amount and remain | | amount  remain |
| **IPO Chart Info** | | **C# Code** | |
| Input: cost  budget  Processing:  Output:  amount  remain  Algorithm:   1. Prompt for cost 2. Accept cost 3. Prompt for budget 4. Accept budget 5. Calculate: amount = budget/cost 6. Calculate: remain = budget-cost\*amount 7. Display amount and remain | | //Input  int cost;  int budget;  //Processing  //Output  int amount;  int remain;  //Algorithm  Console.Write("What’s the cost of the candy: ");  cost = Convert.ToInt32(Console.ReadLine());  Console.Write("How much do you plan to spend on candies: ");  budget = Convert.ToInt32(Console.ReadLine()); amount = budget/cost;  remain = budget-cost\*amount;  Console.WriteLine($"The amount you can buy is {amount}.Your remaining balance is {remain:C}"); | |

1. Thomas’ Towing Services would like an application that calculates and display the cost of a towing job. The cost is based on the towing rate, the distance as well as a flat service rate. The service rate is the same regardless of the towing distance and it is always applied in any job.

Output: cost  
Input: tRate, dist, flatFee  
Sample Calculation:

cost = flatFee + tRate\*dist = 20 + 5\*20 = 120

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Processing** | | **Output** |
| tRate  dist  flatFee | Algorithm:   1. Prompt and accept tRate, dist, flatfee 2. Calculate: cost = flatFee + tRate\*dist 3. Display cost | | cost |
| **IPO Chart Info** | | **C# Code** | |
| Input: tRate  dist  flatFee  Processing:  Output:  cost  Algorithm:   1. Prompt for tRate 2. Accept tRate 3. Prompt for dist 4. Accept dist 5. Prompt for flatFee 6. Accept flatFee 7. Calculate: cost = flatFee+tRate\*dist 8. Display cost | | //Input  int tRate;  int dist;  int flatFee;  //Processing  //Output  int cost;  //Algorithm  Console.Write("What’s the towing rate: ");  tRate = Convert.ToInt32(Console.ReadLine());  Console.Write("How much distance towed: ");  dist = Convert.ToInt32(Console.ReadLine()); Console.Write("What’s the flat fee: ");  flatFee = Convert.ToInt32(Console.ReadLine()); cost = flatFee+tRate\*dist;  Console.WriteLine($"The cost for this towing job is {cost:C}"); | |