

Syed Muhammad Sarim

Lab#2

Lab Tasks:

Q1.

Design a flowchart, Pseudocode, Algorithm for processing a customer order at a restaurant, including handling special requests (Like add on).

Algorithm:

1. Greet the customer by saying **“Welcome, how can I help you?”**
2. Display the **Menu**
3. Ask the customer to enter his **Order**
4. Read the **Order**
5. Ask for any **add on’s**
6. Calculate **Bill**
7. If **add on’s included** then
8. Set **Total Bill** to **Bill + add on bill**
9. Else Set **Total Bill** to **Bill**
10. Get **Payment**
11. Display **Waiting Time**

Pseudo Code:

START

PRINT **“Welcome, how can I help you?”**

DISPLAY **Menu**

PRINT **“Enter your order?”**

INPUT **Order**

PRINT **“Any add on’s?(yes/no)”**

INPUT **add_on’s**

IF **add_on’s = “yes”** THEN

PRINT "Enter add on items"

INPUT addon_items

CALCULATE Total Bill = Bill + Add on Bill

ELSE

CALCULATE Total Bill = Bill

ENDIF

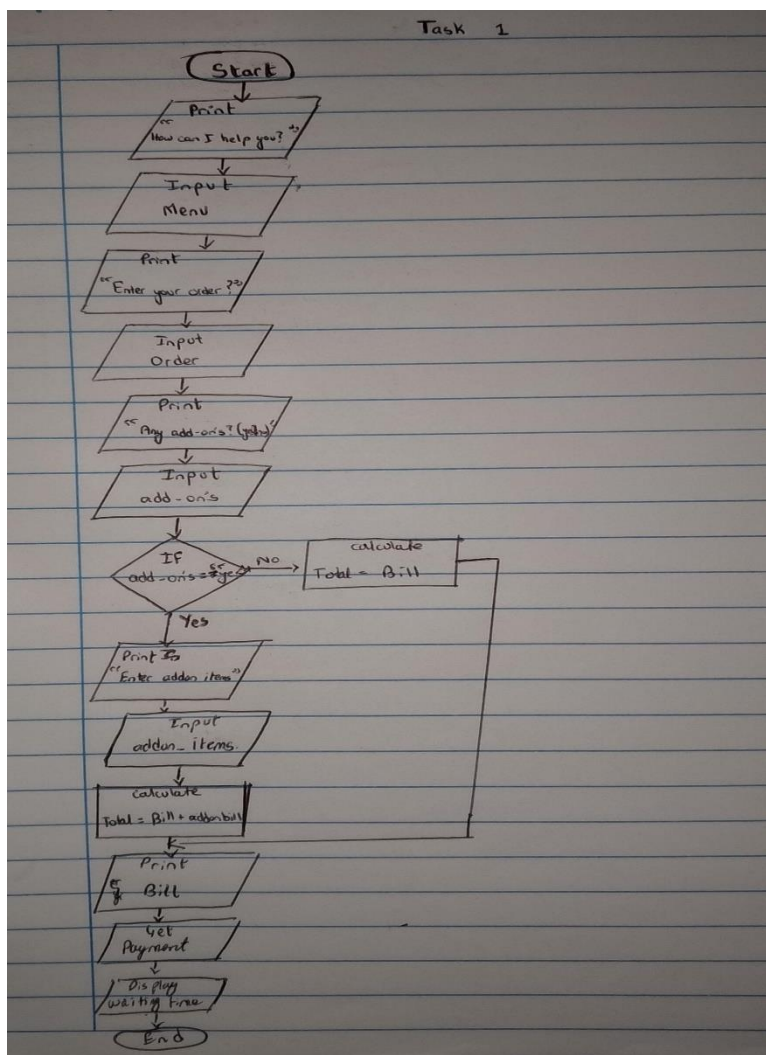
PRINT "Your Bill is", Total Bill

GET Payment

DISPLAY WAITING TIME

END

Flowchart:



Q2.

Design a flowchart, Pseudocode, Algorithm for handling a customer's deposit transaction at a bank, including checks for account validity and deposit amount conditions.

Algorithm:

1. Greet the Customer **"Welcome to ABC Atm"**
2. Ask customer to enter **Atm Card**
2. Read **Atm Card**
3. If **ATM Card** is **Invalid** then
4. Display **"Account Invalid"**
5. Return **Atm Card**
6. Else Display **"How much you want to deposit?"**
7. Read **Amount**
8. If **Amount** is **greater** than **Atm Storage** then
9. Display **"Sorry not enough cash in atm"**
10. Else Give **Amount** and **Receipt**

Pseudo Code:

START

PRINT **" Welcome to ABC Atm and Enter your Atm Card"**

INPUT **Atm Card**

IF **Atm Card = Invalid**

THEN PRINT **"Invalid Account"**

RETURN **Atm Card**

ELSE

PRINT **"Enter Amount?"**

INPUT **Amount**

IF **Amount > Atm Storage**

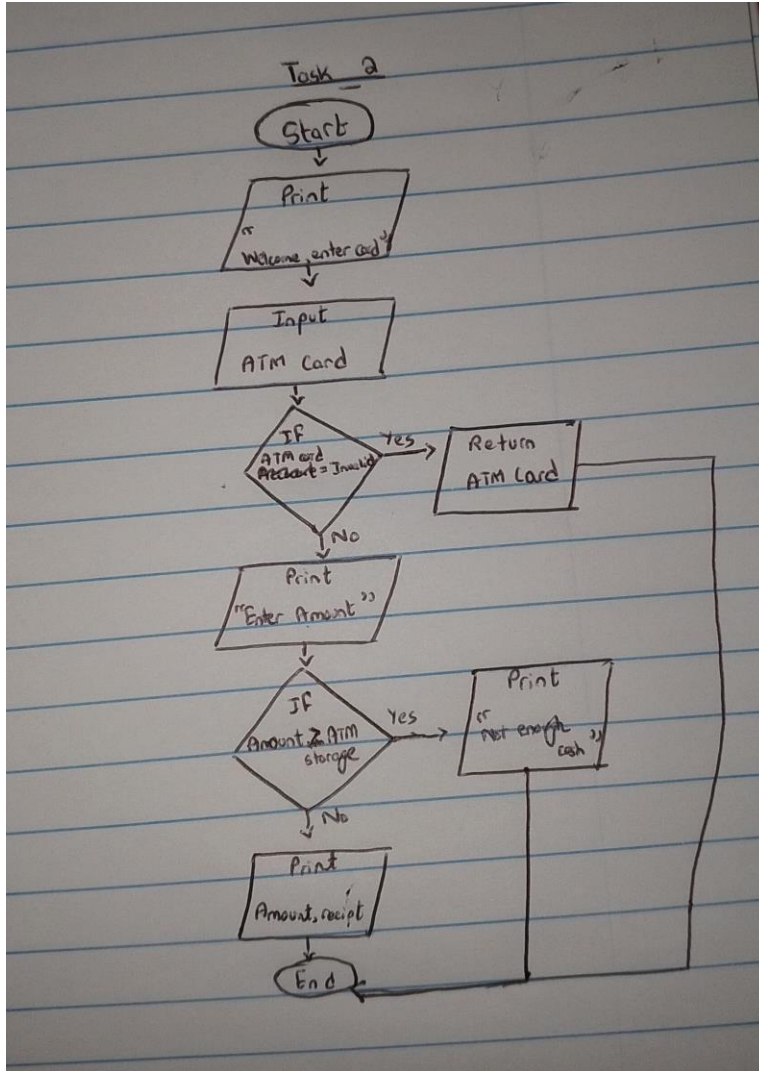
THEN PRINT **" Sorry not enough cash in atm"**

ELSE

PRINT Amount, receipt

END

Flowchart:



Q3.

Design a flowchart, Pseudocode, Algorithm to determine which of three provided numbers is the greatest?

Algorithm:

1. Ask for **Number A,B,C**
2. Read **A ,B and C**

- 3.If **A is greater than B and C** then
- 4.Display **A is greatest**
- 5.Else if **B is greater than A and C** then
- 6.Display **B is the greatest**
- 7.Else if **C is greater than A and B** then
- 8.Display **C is the greatest**
- 9.Else Display **No one number is the greatest**

Pseudo Code:

START

PRINT "Enter num A, B and C"

INPUT A, B, C

IF **A>B and A>C**

THEN PRINT" **A is greatest**"

ELSE IF **B>A and B>C**

THEN PRINT" **B is greatest**"

ELSE IF **C>A and C>B**

THEN PRINT"**C is greatest**"

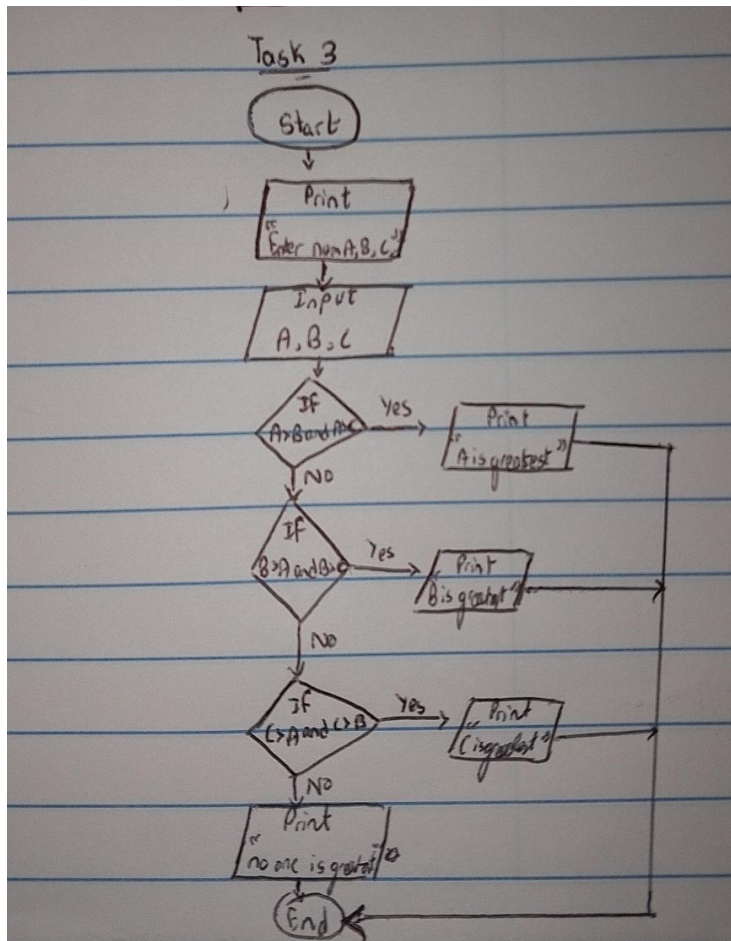
ELSE

PRINT "**No one is the greatest**"

ENDIF

END

FlowChart:



Q4.

Implement an algorithm where the user enters a number, and an appropriate month is displayed.

Algorithm:

1. Ask the user to enter a **number** between **1** and **12**
2. If **number** is **less than 1** or **greater than 12** then display **"incorrect number"**
3. else if **number == 1** then display **January**
4. else if **number == 2** then display **February**
5. else if **number == 3** then display **March**

6. else if **number == 4** then display **April**
7. else if **number == 5** then display **May**
8. else if **number == 6** then display **June**
9. else if **number == 7** then display **July**
10. else if **number == 8** then display **August**
11. else if **number == 9** then display **September**
12. else if **number == 10** then display **October**
13. else if **number == 11** then display **November**
14. else if **number == 12** then display **December**

Q5.

Create pseudocode a small calculator which only does '+' or '-' Operations. (Hint: Take three variable inputs with one being used for the operator)?

Pseudo Code:

START

PRINT "Enter number 1"

INPUT num_1

PRINT "Enter number 2 "

INPUT num_2

PRINT "Enter your operation + or - "

INPUT operator

IF operator == + THEN

Result = num_1 + num_2

ELSE IF operator == - THEN

Result = num_1 - num_2

ELSE

PRINT "Invalid operator. Please enter either '+' or '-'."

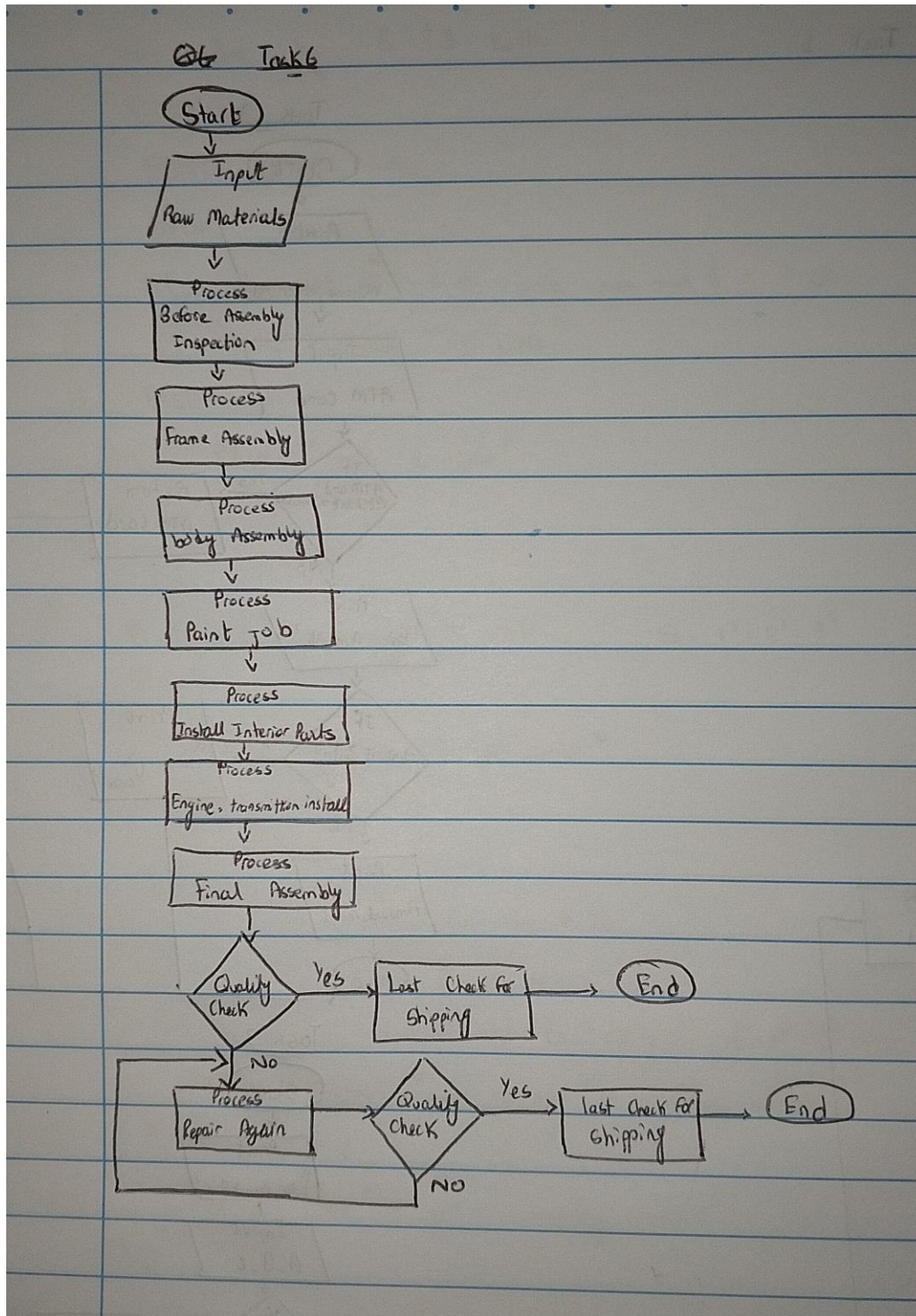
ENDIF

END

Q6.

You are working at Toyota Indus Motors and want to assemble a car. Design a flowchart with proper process modules and decision structures to replicate a pipeline production.

Flowchart:



Q7.

Implement an algorithm for making a simple calculator with all the operators (+, -, *, /, %)

Algorithm:

1. Ask the user for **number 1**, **number 2** and the **operator** (+, -, *, /, %)
2. Read **num_1**, **num_2**, **operator**
3. If **operator** is + Then **Result** is **num_1 + num_2**
4. Else If **operator** is - Then **Result** is **num_1 - num_2**
5. Else If **operator** is * Then **Result** is **num_1 * num_2**
6. Else If **num_2 not equal to 0** and **operator** is / Then **Result** is **num_1 / num_2**
7. Else If **num_2 not equal to 0** and **operator** is % Then **Result** is **num_1 % num_2**
8. Else display **invalid operator**
9. Display **Result**

Q8.

Create your repository with your roll number being your repo name, Upload the algorithms and pseudo codes in your repository, Create a small intro about yourself in the readme file with pictures and bullet points.

https://github.com/sarim8133/Sarim_24k-0718/blob/main/README.md

Q9.

Why we use .gitignore?

Exclude Unnecessary Files: Ignore files that are not needed in the repository, such as temporary files, build artifacts, and logs.

Protect Sensitive Information: Prevent sensitive files, like configuration files with passwords or API keys, from being tracked or shared.

Reduce Clutter: Keep the repository clean by avoiding the inclusion of files that are generated locally and not relevant to other contributors

Q10.

Difference between Algorithm and Pseudocode?

Algorithm	Pseudo Code
An algorithm is a systematic, logical approach that provides a step-by-step procedure for computers to solve a specific problem.	Pseudocode is a simplified version of programming codes, written in plain English language and used to outline a program before its implementation.
Algorithms can be expressed using flowcharts, natural language, and other methods.	Pseudocode includes various control structures such as repeat-until, if-then-else, while, for, and case.