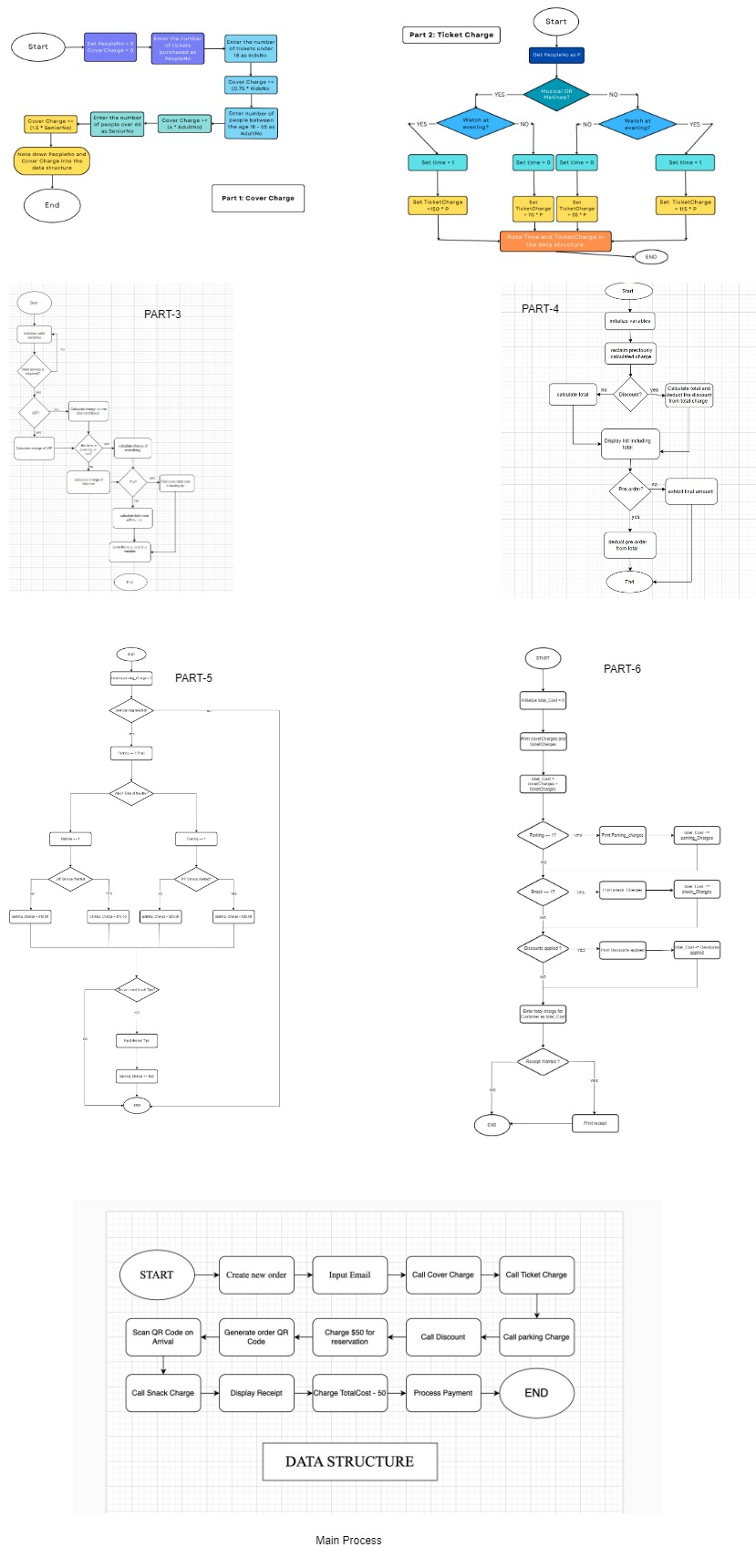
FLOWCHART:

****

FLOWCHART (One Drive Link):[groupflowchart .jpg](https://1drv.ms/i/c/b5c4adf627f3ab1f/ETxEsd1rbg9Jg7rZ9rhJB3QBu5zFDuuIQjW8YIKGpPAhsg?e=S8JiMc)

**Group Pseudo Code:**

**Data Structures:**

- total = 0.00

- subTotal = 0.00 | tax = 0.00

- group\_size | total\_cvrChrg = 0.00

- num\_children = 0 | cvrChrg\_children = 0.00

- num\_adult = 0 | cvrChrg\_adult = 0.00

- num\_senior = 0 | cvrChrg\_senior = 0.00

- showTime = ' '

- showType = ' '

- ticketCharges = 0.00

- discount = 0.00

- final\_ticketCharegs = 0

- valet\_code = 0 | valet\_charges = 0.00

- valet\_choice = ' '

- tip = 0

- snackType = 0 (Non-Alcoholic Beverage = 1, Alcoholic Beverage = 2, Candy = 3)

- snack\_Qty = 0

- cart\_total = 0

**MAIN PROCESS :**

1. START

2. Display : A $50.00 deposit will be charged to secure the reservation.

3.Display : Press "Pay" or "Cancel".

4. If user select "Pay==1" :

display : Proceed to the next step.

5. If user select "Cancel==1" :

Display : Payment Cancelled

6.Genrate {QR code} if payment is succesfullt completed.

7.Display: Your reservation is secured and print the genrated QR code.

8. Call Cover Charge sub process.

9.Call Ticket Charges Sub process

10. Display : Would you like to use the Valet Services.

11.If the user chooses "YES":

- Call Valet : Sub - Process

12. Call Snack bar Sub process

13.[total\_cvrChrg] + [final\_ticketCharegs] + [valet\_charges] + [cart\_total] = [subTotal]

14 . 0.13\* [subTotal] = [tax]

15. ([subTotal] + [tax]) - 50.00 = [total]

16. Call <Receipt : Sub - Process>

17. END

**Cover Charge Sub Process:**

1. START

2. Display:

How many children in your group?

3. Get [num\_children]

4. Display:

How many adults in your group?

5. Get [num\_adult]

6. Display:

How many senior in your group?

7. Get [num\_senior]

8. [num\_children] \* 0.75 = [cvrChrg\_children]

9. [num\_adult] \* 4.00 = [cvrChrg\_adult]

10. [num\_senior] \* 1.50 = [cvrChrg\_senior]

11. [total\_cvrChrg] = [cvrChrg\_children] + [cvrChrg\_adult] + [cvrChrg\_senior]

12. [group\_size] = [num\_children] + [num\_adult] + [num\_children]

13. END

**Ticket Charges: Sub-Process**

1. Start

2.Display: Which show you want to watch

-Matinée

- Evening

3.Input [showTime]

4. Display: Which show you want to watch?

- Musical

- Theatre

5.Input the showType.

6. If the [showTime] is "Matinée"

- [valet\_code] = 1

What is the show type?

(a) If [showType] is "Musical":

- 70 \* [group\_size] = [ticketCharges]

(b) If [showType] is "Theatre":

- 55 \* [group\_size] = [ticketCharges]

7. If the [showTime] is "Evening"

- [valet\_code] = 2

What is the show type?

(a) If [showType] is "Musical":

- 150 \* [group\_size] = [ticketCharges]

(b) If [showType] is "Theatre":

- 155 \* [group\_size] = [ticketCharges]

8.END

**Discounts: Sub process**

1. Start

2.Is [group\_size] equal to 2?

(a) (YES):

- 0.05 \* [ticketCharges] = [discount]

(b) (NO):

- Proceed to the next step

3. Is [group\_size] more than 2?

(a) (YES):

Is [group\_size] equal 3?

- (YES):

- 0.15 \* [ticketCharges] = [discount]

- (NO):

- Proceed to the next step

4. Is [group\_size] more than 3?

(a) (YES):

Is [group\_size] equal 4?

- (YES):

- 0.20 \* [ticketCharges] = [discount]

- (NO):

- Proceed to the next step

5. Is [group\_size] more than 4?

(a) (YES):

Is [group\_size] equal 5 or more?

- (YES):

- 0.30 \* [ticketCharges] = [discount]

- (NO):

Proceed to the next step

6. Is [group\_size] less than 2?

(a) YES:

0.00 \* [ticketCharges] = [discount]

7. [ticketCharges] - [discount] = [final\_ticketCharegs]

8. END

**Valet : Sub Process**

1. Start

2.Display : Do you want the standard parking or VIP?

3. Did the user select "Standard"?

- [valet\_choice] = 'Standard'

(a) (YES):

Is [valet\_code] = 1?

(YES):

[valet\_charges] += 10.00

(NO):

[valet\_charges] += 20.00

4. Did the user select "VIP"?

- [valet\_choice] = 'VIP'

(a) (YES):

Is [valet\_code] = 1?

(YES):

[valet\_charges] += 15.00

(NO):

[valet\_charges] += 30.00

5. Display: Do you want to give tip ?

6. If the user chooses "NO"

- Proceed to the next step

6.. If the user choose "YES"

Display:

"How much would you like to tip?: "

7. Get [tip]

8.[valet\_charges] += [tip]

9. END

**Snack Bar: Sub Process**

1. START

2. Display : Snack Prices

-Non-Alcoholic Beverages - 2.25

-Alcoholic Beverage - 7.50

-Misc. Candy Item - 1.75

3. Display:

Which Snack would you like to eat?

1. Non-Alcoholic Beverage

2. Alcoholic Beverage

3. Misc. Candy Item

4. No Snacks

4. Did the user choose "Non-Alcoholic Beverage"?

(YES):

- [snackType] = 1

(NO):

- Proceed to the next step

5. Did the user choose "Alcoholic Beverage"?

(YES):

- [snackType] = 2

(NO):

- Proceed to the next step

6. Did the user choose "Misc. Candy Item"?

(YES):

- [snackType] = 3

7. Is the [snackType] equal to 1?

(YES):

Display:

" How many Non-Alcoholic Beverage would you like? : "

(NO):

Is the [snackType] equal to 2?

(YES):

Display :

" How many Alcoholic Beverages would you like? : "

(NO):

Is the [snackType] equal to 3?

(YES):

Display :

" How many Misc. Candy Item would you like? : "

8. Get [snack\_Qty]

9. Is [snackType] equal to 1?

(YES):

- cart\_total += 2.25 \* [snack\_Qty]

(NO):

Is [snackType] equal to 2?

(YES):

- cart\_total += 7.50 \* [snack\_Qty]

(NO):

Is [snackType] equal to 3?

(YES):

- cart\_total += 1.75 \* [snack\_Qty]

10. Display: Do you wish to get any other snacks?

11. If user selected "YES":

- Go back to step 3

12. END

**Receipt : Sub - Process**

1.START

2.Display :

A La Carte Entertainment

Cover Charges :

Cover charges For Children = cvrChrg\_children

Cover charges for adults = [cvrChrg\_adult]

Cover charges for seniors = [cvrChrg\_senior]

total\_cvrChrg

Ticket Charges:

[showTime] [showType] = [ticketCharges]

Discounts = [discount]

[final\_ticketCharegs]

Snack Charges = [cart\_total]

Valet Charges:

[showTime] [valet\_choice] = [valet\_charges]

(TIP - [tip])

Sub-Total = [subTotal]

Tax (13%) = [tax]

Total (-$50.00 of deposit) =[total]

3. END

**Test Scenario 1 :**

- 2 adults, 1 child, 2 seniors plan to go to the venue at 11:30 am and see musical

- deposit: $50.00

- cover charge: $11.75

- ticket: $350.00

- discount: 30% off

- total ticket charge: $245.00

- They opt for the VIP valet service

- valet charge: $10.00

- At intermission, they order one bag of chips, three Sprite pops, and one wine

- snack charge: $16.00

- After the production is over, they tip $4.00 for valet parking

- total valet charge: $14.00

- Subtotal: $286.75

- Tax: $37.28

- Total: $274.03

- Get receipt

- They made full payment and leave the building

**Test Scenario 2:**

- 2 adults plan to go to the venue at 7:00 pm and see the theatre production

- deposit: $50.00

- cover charge: $8.00

- ticket: $230.00

- discount: 5% off

- total ticket charge: $218.50

- They opt for the Standard valet service

- valet charge: $20.00

- At intermission, they order two beers and one plate of nachos

- snack charge: $16.75

- After the production is over, they don't tip for valet parking

- Subtotal: $263.25

- Tax: $34.22

- Total: $247.47

- Get receipt

- They made full payment and leave the building.

**Computational Thinking Explanation**

**Pseudocode Explanation:**

The combined pseudocode for the Workshop integrates 3 logical parts (1, 2, 3) to create cohesive application for managing fees and offerings at A La Carte Entertainment. Each element addresses precise components of the problem, such as calculating cover charges, ticket fees, discounts, snack purchases, and valet parking charges. By breaking down the problem into smaller components and identifying the essential inputs, calculations, and outputs, the pseudocode ensures clarity and efficiency in the development process.

Computational thinking principles guide the design of the pseudocode by emphasizing abstraction, decomposition, pattern recognition, and algorithm design. Abstraction allows us to focus on critical factors while ignoring irrelevant details, ensuring that the pseudocode remains concise and easy to understand. Decomposition involves breaking down the process into smaller, manageable tasks, making it easier to develop algorithms for each component. Algorithm design ensures that the pseudocode provides step-by-step instructions for performing tasks accurately and efficiently. Overall, the combined pseudocode offers a comprehensive solution that addresses all requirements of the application while adhering to computational thinking principles.

**Flowchart Explanation:**

The combined flowchart for the Workshop visually represents the entire application process, integrating the individual flowcharts for each logic (1, 2, 3) into a cohesive structure. Each flowchart depicts the sequence of steps and decision points involved in managing fees, charges, discounts, and services at A La Carte Entertainment, providing a clear visual representation of the application's functionality.

The flowchart begins with an initial decision point where the user chooses whether to calculate charges, ticket fees, or discounts. From there, the flowchart branches into three main paths, corresponding to each logical component. Each path includes decision points, process steps, and terminal points that guide the user through the application process.

By combining the individual flowcharts, the combined flowchart effectively illustrates the flow of information and decision-making in the application. It provides a visual roadmap for navigating the different components of Workshop 5, ensuring clarity and coherence in the overall design. Overall, the combined flowchart offers a comprehensive overview of the application's functionality while adhering to computational thinking principles.