

HOLY ANGEL UNIVERSITY**School of Computing****6LOGPROG****Group No: 8****Date: November 09, 2022****Group Members: Borja, Karl Vladimir B.
Cayanan, Luis Miguel I.
Paldeng, Carl Lester P.
Quiambao, Mark Erolle****Topic: Catering Services****Program Description**

- BURI acts as a program where consumers can select products and order catering services. It is a program that receives user inputs for operations, orders and user information for order processing. This application also has built-in calculations for user orders, including quantity, costs for each order based on quantity, total purchase for each section, reset cart and overall total. It prompts the user input and display its specific function. If a user's input is not recognized, it asks for a valid input. The program provides various catering options for users to browse, compare, purchase, display total purchase, and check out.

Purpose

- Secured order transaction
- Saves time, money and effort
- Ease of Access
- Less physical contact
- No pressure on ordering
- View and compare products/services offered
- Convenient shopping

Major Features

- Options for each dedicated menu
- Input validation (display specific functions according to user input such as viewing of main options, carts, packages, order, quantity etc.)
- If user input is not recognized, it reminds the users and prompts for a valid input
- Built in calculation for user orders such as orders and their corresponding total based on their quantity, total of each section (party food, rentals and party needs, catering services) and overall total
- Provide a main cart which provides the list of items ordered from different sections along with their quantity, price, an option to checkout, reset all orders, return to main menu and total price.
- Obtain user information and details for order processing and display total amount, total to pay and various information.

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

Program Flow

Program Guide

By the time you open the program, you are presented with 6 options, each with their own dedicated menu and functions. The program will then ask for your input.

```

C:\Windows\py.exe
0=====***BURI CATERING SERVICE***=====0
                                [1] Read More
                                [2] Party Food
                                [3] Rentals and Party Needs
                                [4] Catering Services
                                [5] View your cart
                                [0] End Program
0=====0
Input Operation[0-5]: 
  
```

To order food, the user must simply choose within the given options. For example, the options in Party Food. The program also asks if the user wants to order or go back to the previous page.

```

0=====0
Input Operation[0-5]: 2
0=====***PARTY PACK OPTIONS***=====0
                                [1]Packs
                                [2]Cart
                                [3]Back
0=====0
Input Operation[1-3]: 1
0=====0
| PACKAGE CODE [1] | PACKAGE CODE [2] | PACKAGE CODE [3] |
| Package 1: Occasional Package | Package 2: Party Package Overload | Package 3: Formal Package |
| (Good for 20 People) | (Good for 20 People) | (Good for 20 People) |
| Mixed Veggies and Seafood | Fried Garlic Rice | Roast Beef |
| Spaghetti & Meatballs | Hotdogs | Fried garlic and butter rice |
| Plain Rice | Crispy Chicken Tacos | Chicken Cordon Bleu |
| Garlic Chicken | Pizza Hawaiian | Roast Pork with Oyster Sauce |
| Roast Beef | Chicken Wings | Pescada en Salsa Verde |
| Pork and Chicken BBQ | BBQ Pork Sliders | Baked Spaghetti |
| Fish Fillet | Spaghetti & Meatballs | Mixed Vegetables with Ham |
| Pork Lumpia | Sandwiches Roasted Beef | Buko Fruit Salad |
| Iced Tea Bottomless | Iced Tea Bottomless | Iced Tea Bottomless |
| Ice Mineral Water | Ice Mineral Water | Ice Mineral Water |
| Price: P 10,261 | Price: P 11,707 | Price: P 12,404 |
0=====0
                                [0] Order    [B] Back
Input Operation[0/B]: 0
  
```

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

If the user wants to order, he/she is then asked to input a valid package code and quantity.

```

0=====0
| PACKAGE CODE [1] | PACKAGE CODE [2] | PACKAGE CODE [3] |
| Package 1: Occasional Package | Package 2: Party Package Overload | Package 3: Formal Package |
| (Good for 20 People) | (Good for 20 People) | (Good for 20 People) |
| Mixed Veggies and Seafood | Fried Garlic Rice | Roast Beef |
| Spaghetti & Meatballs | Hotdogs | Fried garlic and butter rice |
| Plain Rice | Crispy Chicken Tacos | Chicken Cordon Bleu |
| Garlic Chicken | Pizza Hawaiian | Roast Pork with Oyster Sauce |
| Roast Beef | Chicken Wings | Pescada en Salsa Verde |
| Pork and Chicken BBQ | BBQ Pork Sliders | Baked Spaghetti |
| Fish Fillet | Spaghetti & Meatballs | Mixed Vegetables with Ham |
| Pork Lumpia | Sandwiches Roasted Beef | Buko Fruit Salad |
| Iced Tea Bottomless | Iced Tea Bottomless | Iced Tea Bottomless |
| Ice Mineral Water | Ice Mineral Water | Ice Mineral Water |
| Price: P 10,261 | Price: P 11,707 | Price: P 12,404 |
0=====0
| [0] Order | [B] Back |
Input Operation[0/B]: 0
Input Pack Code [1-3]: 1
Quantity: 1

| [Successfully added to your cart!!!] |
0=====0
| ***PARTY PACK OPTIONS*** |
| [1]Packs |
| [2]Cart |
| [3]Back |
0=====0
Input Operation[1-3]:

```

If the user wishes to view the cart, the items ordered will be provided. It also asks if the user wants to return to the previous page or reset his/her cart. If user wants to reset the cart, it is cleared and will display no current items if viewed again.

```

0=====0
| [1]Packs |
| [2]Cart |
| [3]Back |
0=====0
Input Operation[1-3]: 2
| ***Party Food Cart*** |
| Package Ordered | Quantity | Price |
| Party Package 1 [Occasional Package]: | 1 | P 10,261.00 |
| | | Party Pack Total: P 10,261.00 |
| [R] Reset Order [B] Back |
Input Operation[R/B]: R
| Successfully Cleared Your Party Cart!!! |
0=====0
| ***PARTY PACK OPTIONS*** |
| [1]Packs |
| [2]Cart |
| [3]Back |
0=====0
Input Operation[1-3]: 2
| ***Party Food Cart*** |
| There are no current items in your shopping cart. |
| [R] Reset Order [B] Back |
Input Operation[R/B]:

```

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

Once the user has placed his/her orders, it would be shown with their order name, quantity, price and total price on the main menu [5] cart.

```

C:\Windows\py.exe
=====0
Input Operation[0-5]: 5
=====0
***Your Cart***=====0

***Party Food Cart***
Package Ordered      Quantity      Price
Party Package 1 [Occasional Package]:      1      P 10,261.00
Party Package 2 [Party Package Overload]:    1      P 11,707.00
Party Package 3 [Formal Package]:           1      P 12,404.00
Party Pack Total: P 34,372.00

***Equipment Rental Cart***
Package Ordered      Quantity      Price
Complete Catering Eq. Rental Package:      1      P 100.00
Complete Catering Eq. Rental Package w/ Waiters: 1      P 130.00
Complete Catering Eq. w/ Flower Arrangement & Waiters: 1      P 160.00
Tents:              50      P 5,000.00
Tables:             50      P 5,000.00
Chairs:             50      P 1,000.00
Table Cloth Linen:  15      P 750.00
Ribbons:            50      P 750.00
Complete Utensils and Silverware:           50      P 2,500.00
Equipment Total: P 15,390.00

***Catering Cart***
Package Ordered      Quantity      Price
Wedding Package 1:    1      P 24,000.00
Birthday Package 2:    1      P 28,000.00
Catering Pack Total: P 52,000.00

=====0
***Total***=====0
Your current total is P 101,762.00

[1]CHECKOUT
[2]RESET ALL ORDERS
[3]BACK
=====0
Input Operation[1-3]:
=====0

```

The program also asks if the user wants to checkout, reset orders or return to previous page. If the user input 2, the program resets all user orders. If the user checkouts, he/she is asked for user details. Once done, the program displays user details for confirmation. The user is also asked if he/she wants to reset the details provided. If so, the program will ask for an updated version of the user details.

```

C:\Windows\py.exe
[1]CHECKOUT
[2]RESET ALL ORDERS
[3]BACK
=====0
Input Operation[1-3]: 1
=====0
Place your details and accurate data. Invalid information will not be processed by the management.
Full Name: Carl Lester Paldeng
Contact Number: 09123456789
Venue Address: Sindalan, Davao
Input date of order (Ex. January 3,2022): November 09, 2022
Input date of order delivery (Ex. January 13, 2022): December 6, 2022
Guest Count: 50
Mode of Payment [GCASH/Paymaya] Gcash
=====0

Full Name: Carl Lester Paldeng
Contact Number: 09123456789
Venue Address: Sindalan, Davao
Date of order: November 09, 2022
Date of order deadline: December 6, 2022
Guest Count: 50
Mode of Payment: Gcash

Total Payment P: 101,762.00

Make sure to print out all details accurately as our management will check the data submitted.
Any form of innacuracy or invalid data will not be processed. Thank you for your understanding.
=====0
Have you confirmed your details?
[1] Yes [2] Reset my details
=====0
Input Operation: 1
=====0

```

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

Once user details have been confirmed, the user total order is displayed, as well as the amount for order processing. The program also asks if the user wants to create another order or not.

```

0=====0
    Thank you Carl Lester Paldeng for placing this order. Your order will be processed and you will
    be contacted by one of our staff within 1-3 business days.

    Your bill came to a total of P 101,762.00 .
    Please prepare 50% of your total bill which is 50,881.00 php
    for the processing of your order.

    Once your order is processed, an agreement will be provided for
    service finalization.

    Thank you for being our valued customer! We'll make sure that you will have a blast!

0=====0
                                Do you want to create another order?
                                [1]Yes  [2] No
0=====0
Input Operation:
  
```

If the user wants to order again, then the program will return to the main options and reset all previous orders.

```

0=====0
                                Do you want to create another order?
                                [1]Yes  [2] No
0=====0
Input Operation: 1
0=====0
                                ***BURI CATERING SERVICE***
                                [1] Read More
                                [2] Party Food
                                [3] Rentals and Party Needs
                                [4] Catering Services
                                [5] View your cart
                                [0] End Program
0=====0
Input Operation[0-5]: 5
                                You have no current orders
0=====0
Press [0] to go back:
  
```

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

In any case the user input is not recognized, the program displays an invalid input and ask for another input.

```

0=====***BURI CATERING SERVICE***=====0
      [1] Read More
      [2] Party Food
      [3] Rentals and Party Needs
      [4] Catering Services
      [5] View your cart
      [0] End Program
0=====0
Input Operation[0-5]: asd
Invalid Input
0=====***BURI CATERING SERVICE***=====0
      [1] Read More
      [2] Party Food
      [3] Rentals and Party Needs
      [4] Catering Services
      [5] View your cart
      [0] End Program
0=====0
Input Operation[0-5]: 2

C:\Windows\py.exe
0=====***PARTY PACK OPTIONS***=====0
      [1]Packs
      [2]Cart
      [3]Back
0=====0
Input Operation[1-3]: 1
0=====0
      PACKAGE CODE [1]      PACKAGE CODE [2]      PACKAGE CODE [3]
      Package 1: Occasional Package | Package 2: Party Package Overload | Package 3: Formal Package
      (Good for 20 People) | (Good for 20 People) | (Good for 20 People)
      Mixed Veggies and Seafood | Fried Garlic Rice | Roast Beef
      Spaghetti & Meatballs | Hotdogs | Fried garlic and butter rice
      Plain Rice | Crispy Chicken Tacos | Chicken Cordon Bleu
      Garlic Chicken | Pizza Hawaiian | Roast Pork with Oyster Sauce
      Roast Beef | Chicken Wings | Pescada en Salsa Verde
      Pork and Chicken BBQ | BBQ Pork Sliders | Baked Spaghetti
      Fish Fillet | Spaghetti & Meatballs | Mixed Vegetables with Ham
      Pork Lumpia | Sandwiches Roasted Beef | Buko Fruit Salad
      Iced Tea Bottomless | Iced Tea Bottomless | Iced Tea Bottomless
      Ice Mineral Water | Ice Mineral Water | Ice Mineral Water
      Price: P 10,261 | Price: P 11,707 | Price: P 12,404
0=====0
      [0] Order      [B] Back
Input Operation[0/B]: 0
Input Pack Code [1-3]: 4
Invalid Input
Input Pack Code [1-3]: asd
Invalid Input
Input Pack Code [1-3]: 1
Quantity: 0
Input must be greater than 0
Quantity: -1
Input must be greater than 0
Quantity: _

0=====0
      Have you confirmed your details?
      [1] Yes [2] Reset my details
0=====0
Input Operation: no
Invalid Option
Input Operation: _

```

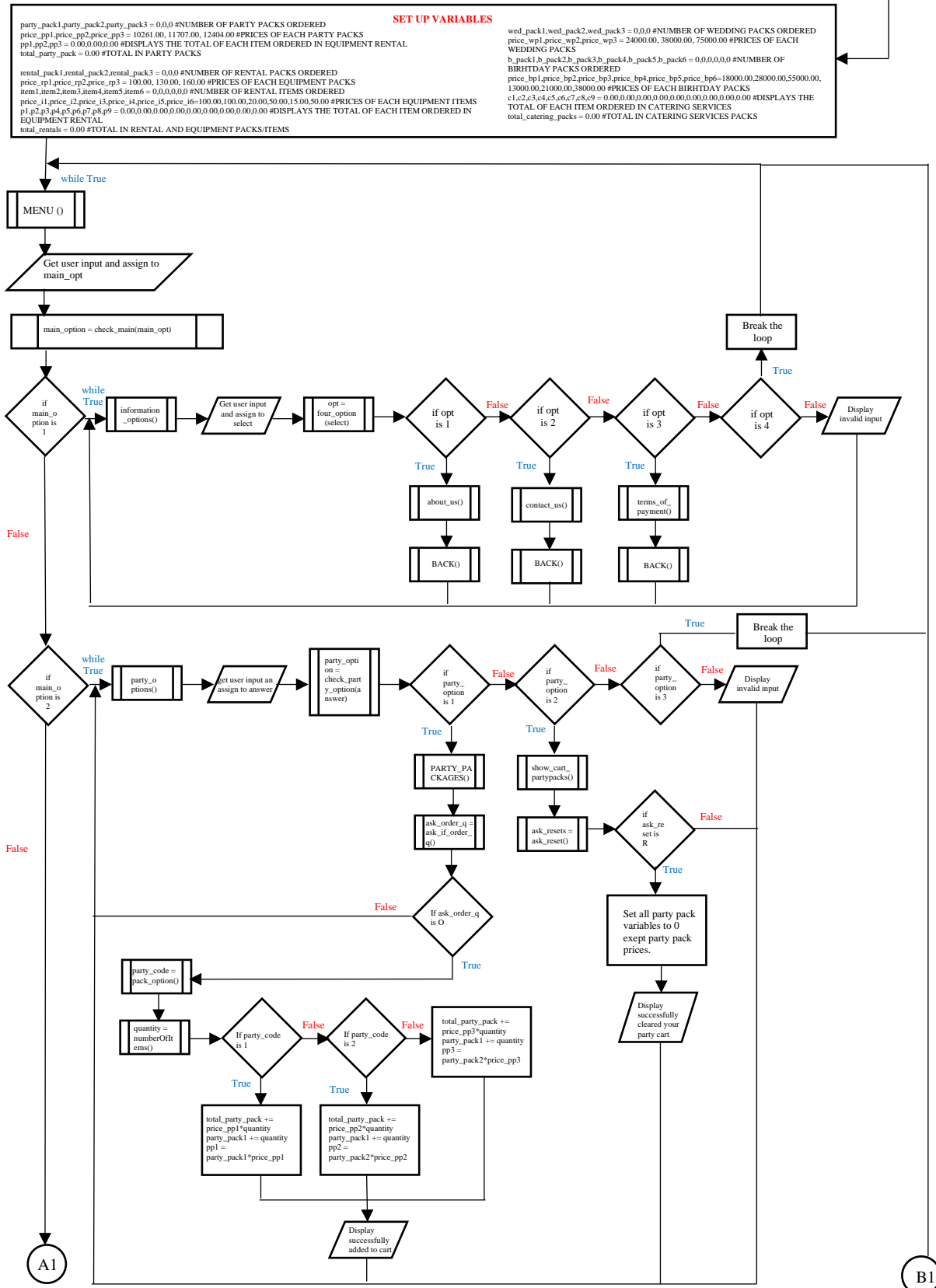
In addition, if the user wants to end the program, he/she must simply input the assigned value to end the program at the right section.

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

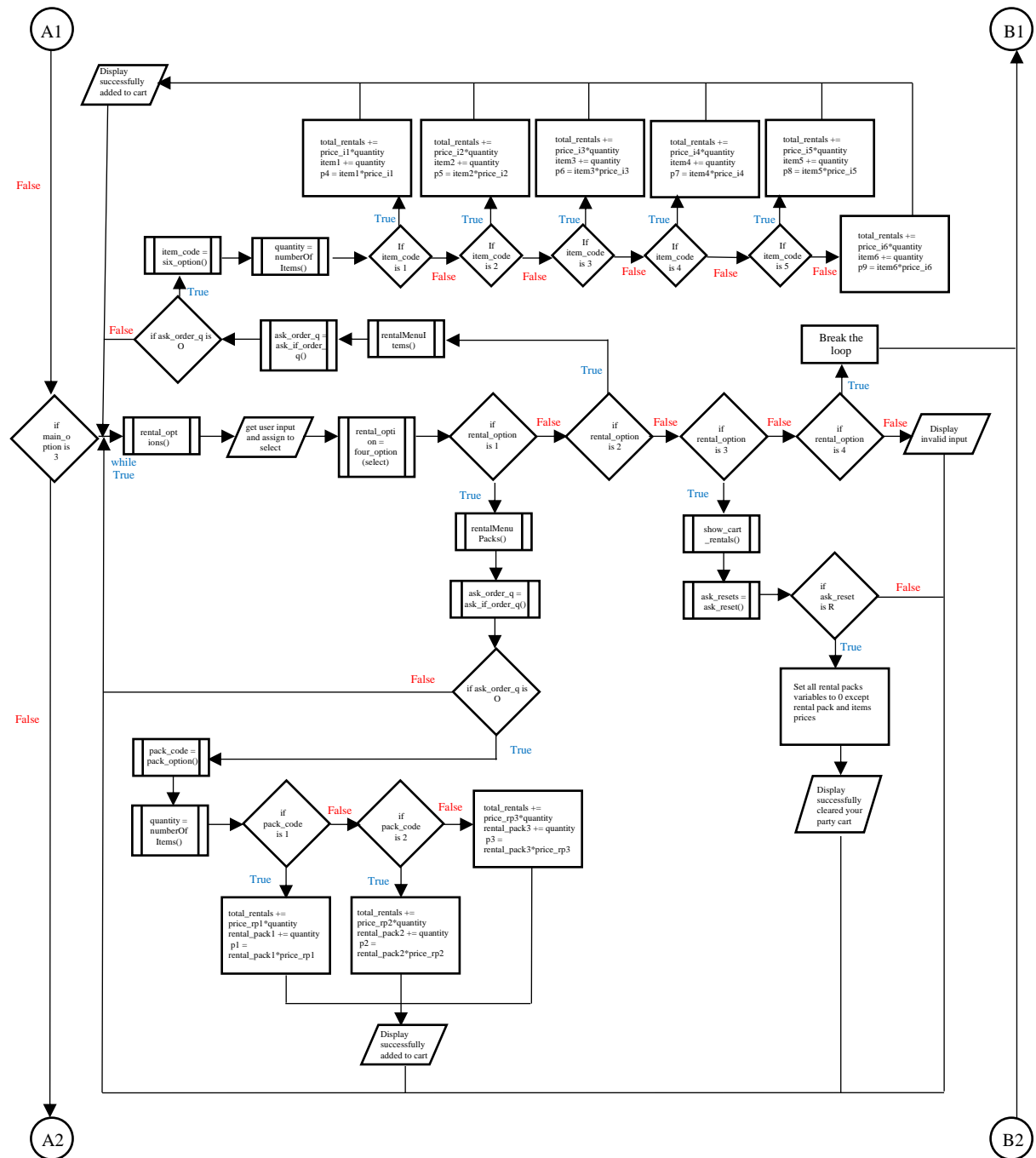
Program Flowchart



HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

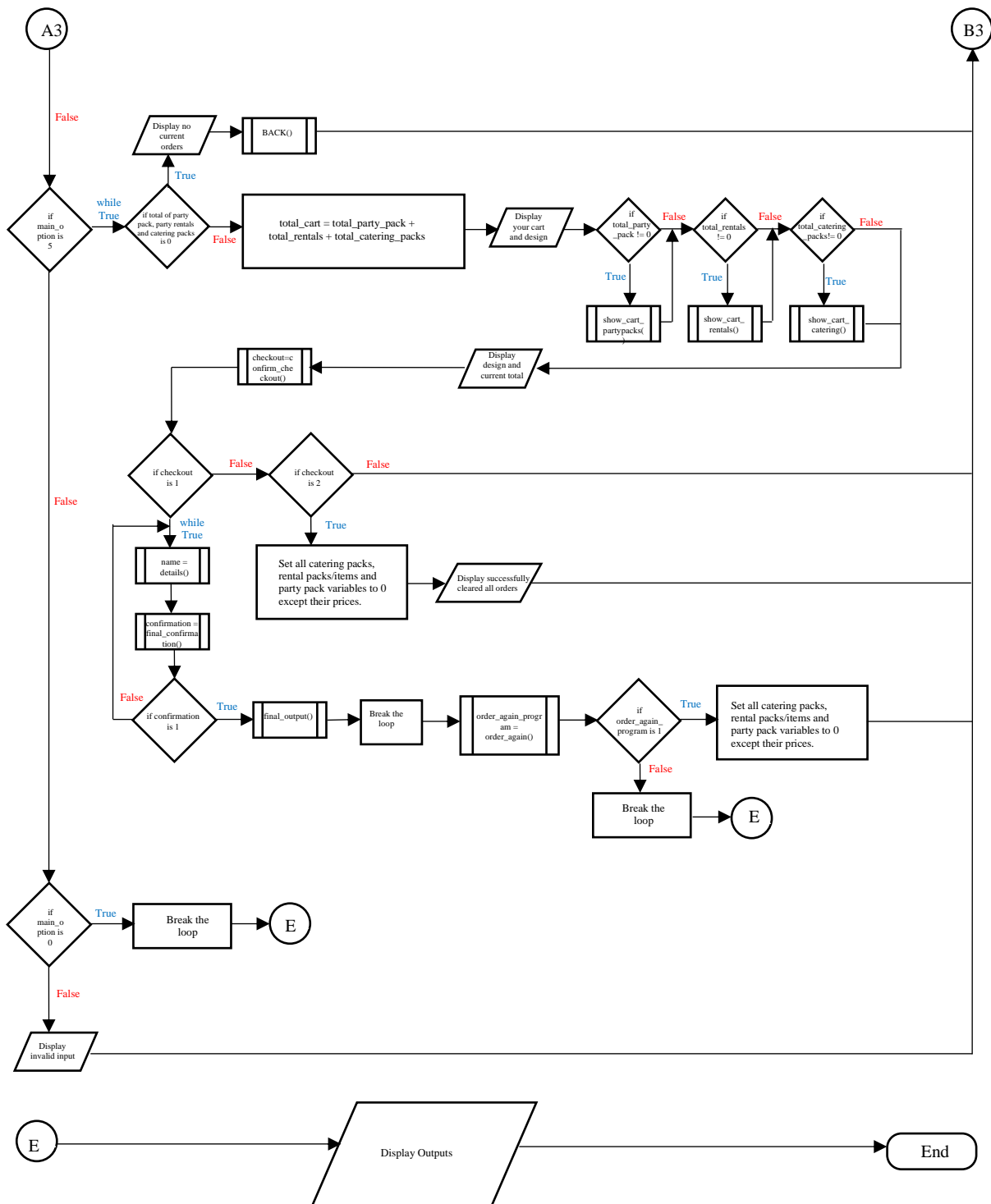




HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG



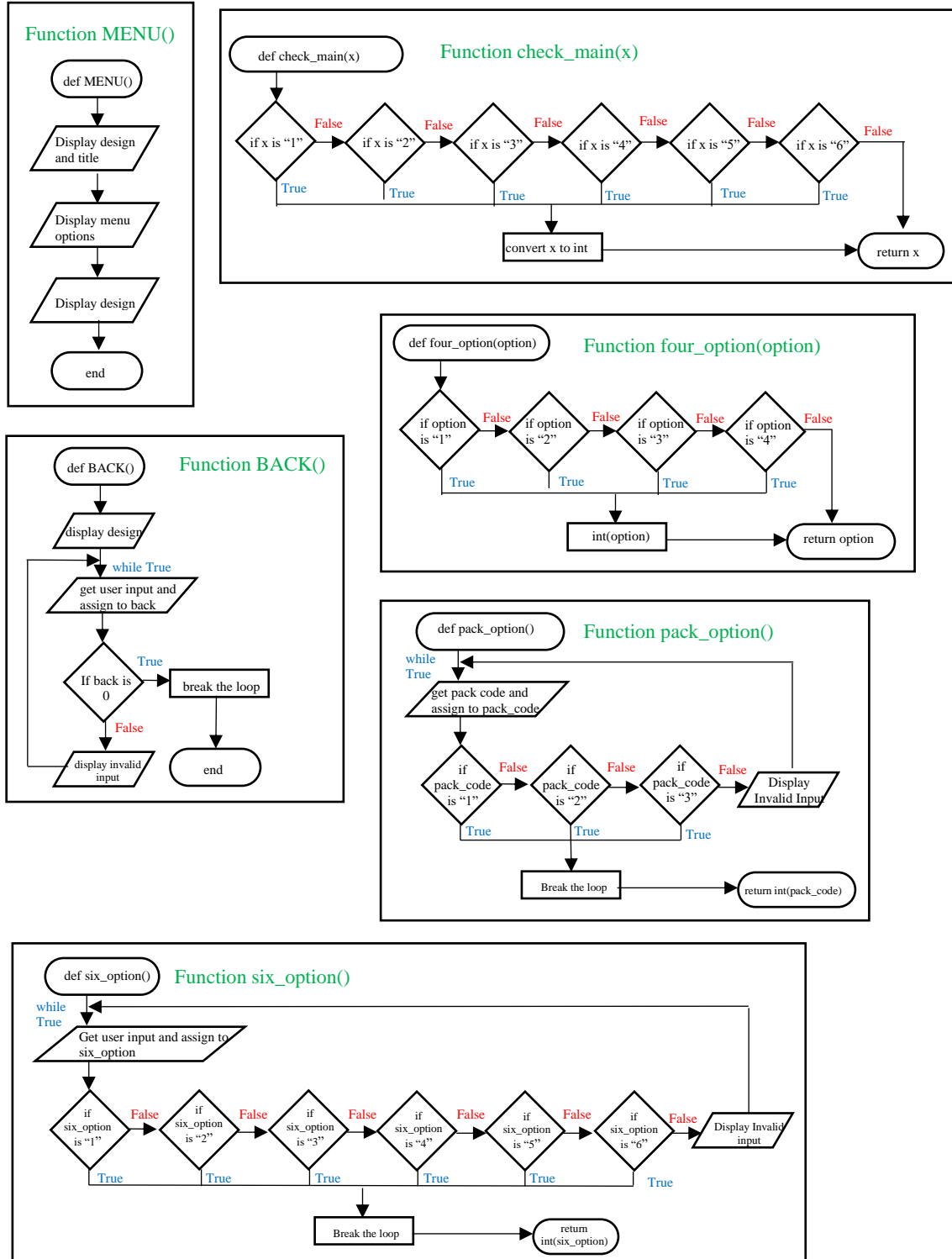
HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

Flowchart for Functions

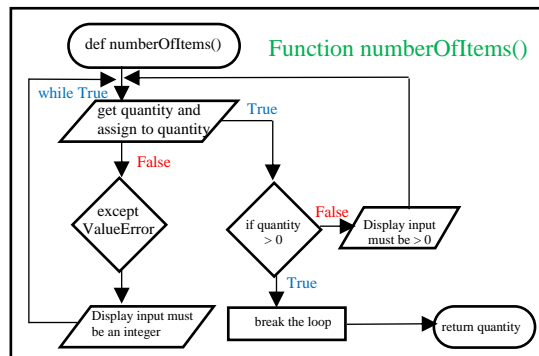
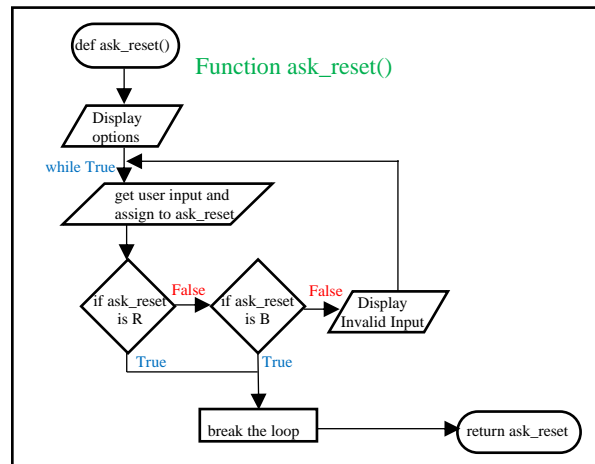
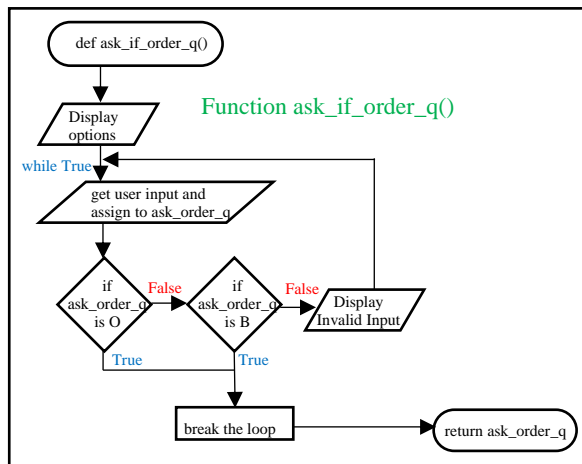
Main Functions



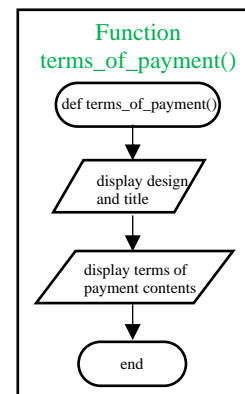
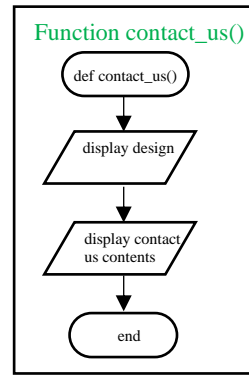
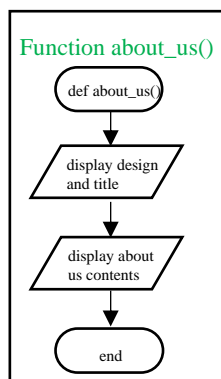
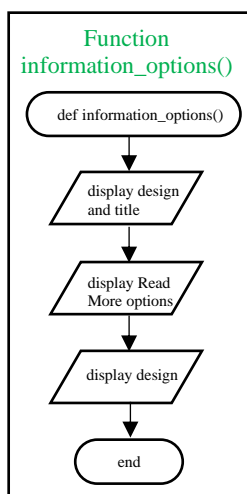
HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG



Read More Functions

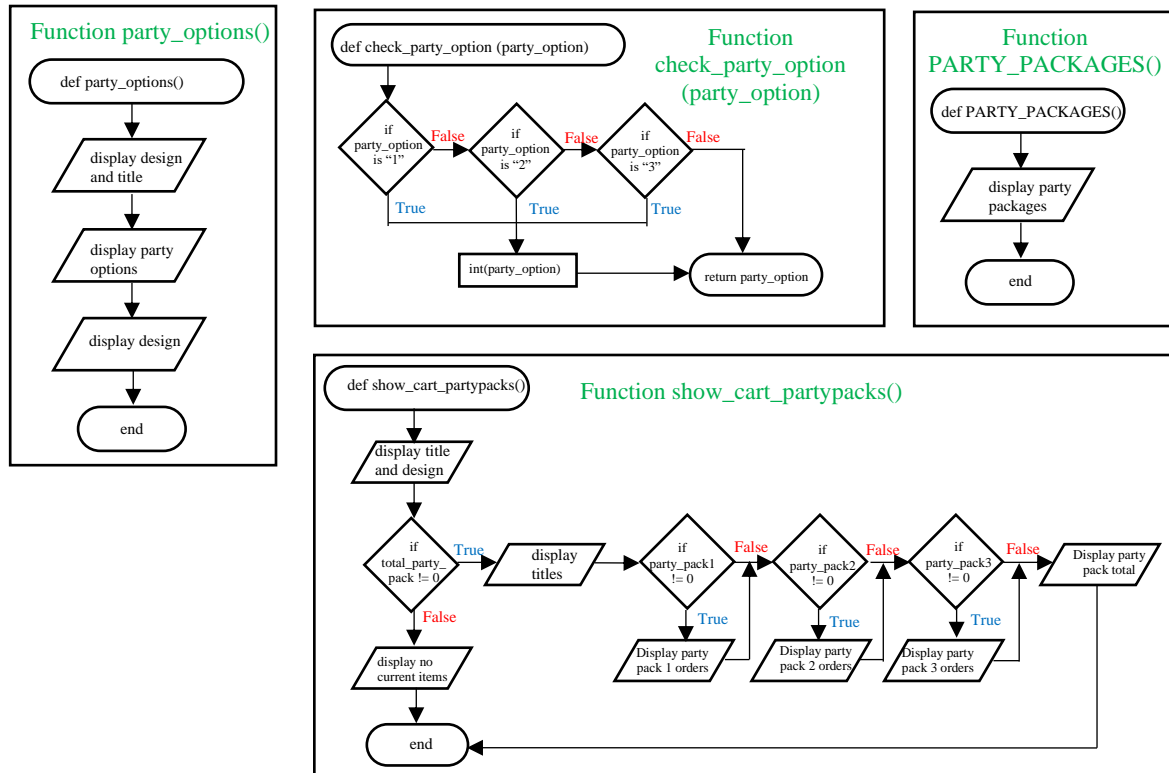


HOLY ANGEL UNIVERSITY

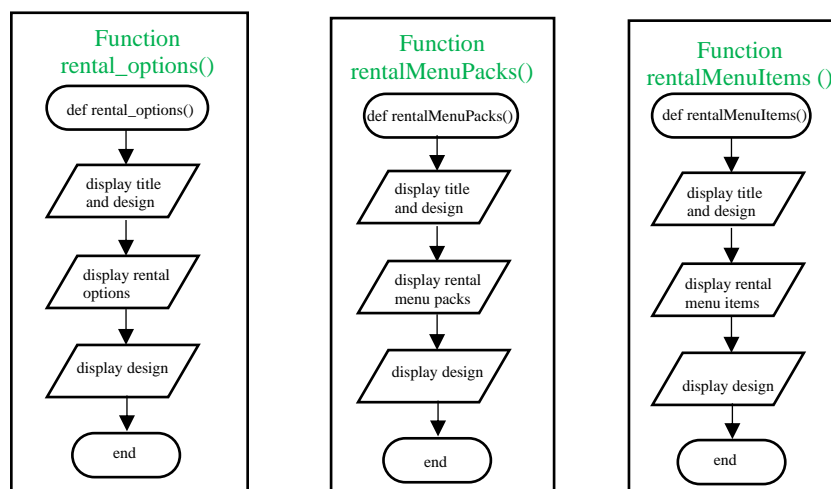
School of Computing

6LOGPROG

Party Food Functions



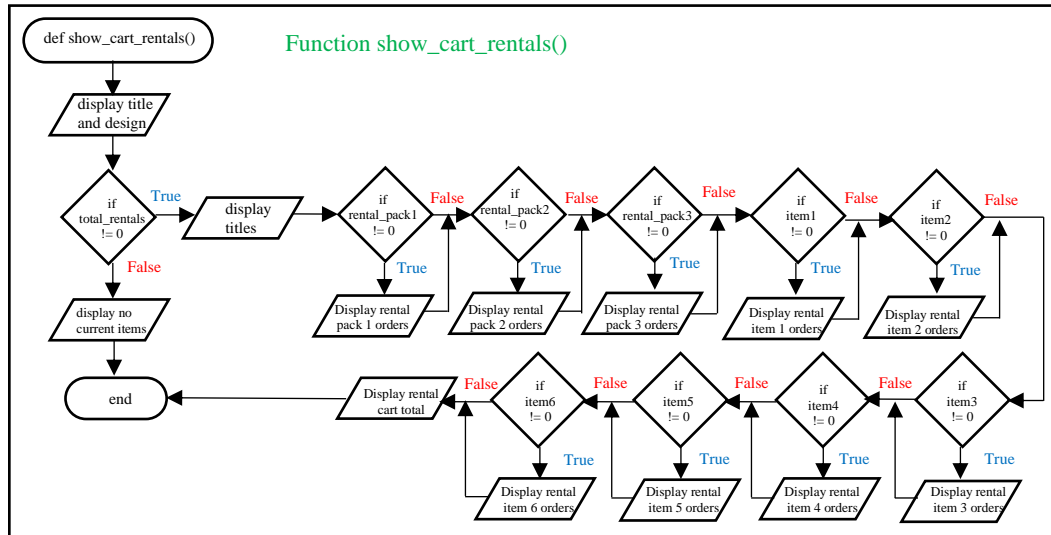
Rentals and Party Needs Functions



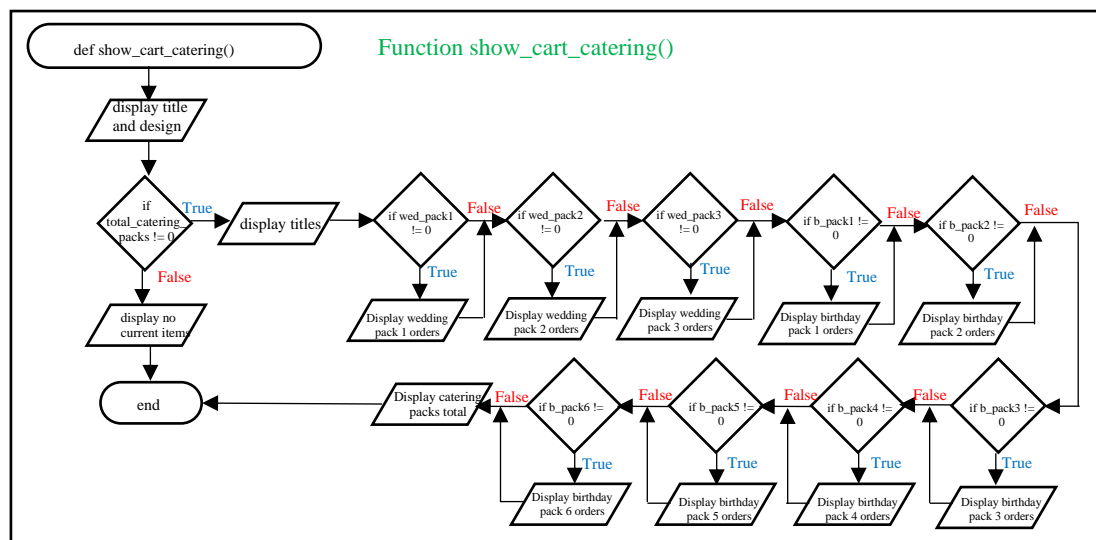
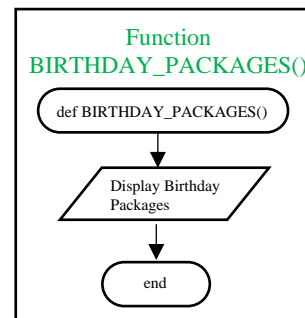
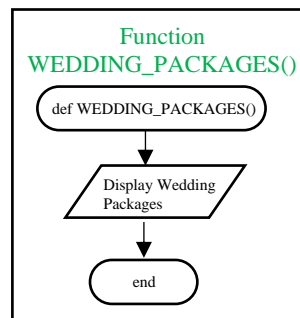
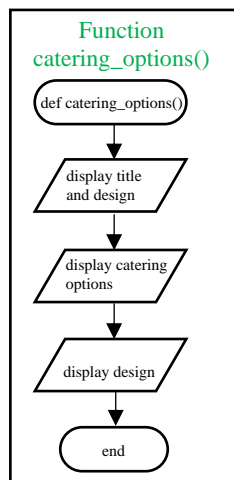
HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG



Catering Services Functions

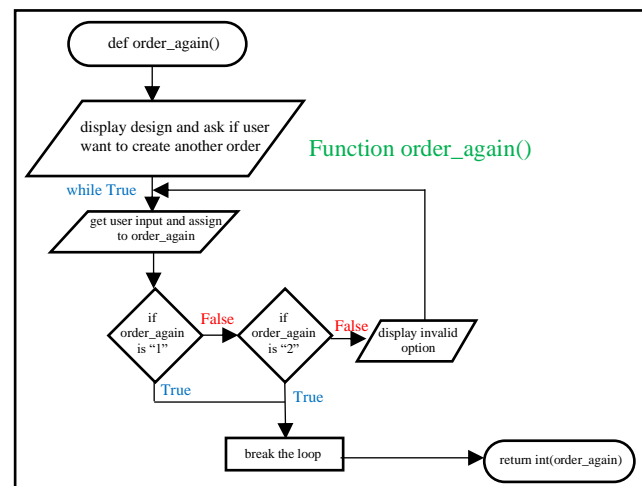
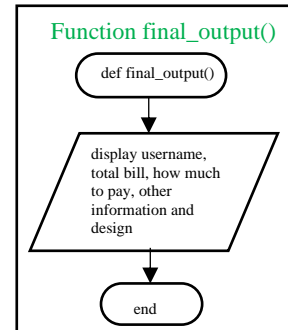
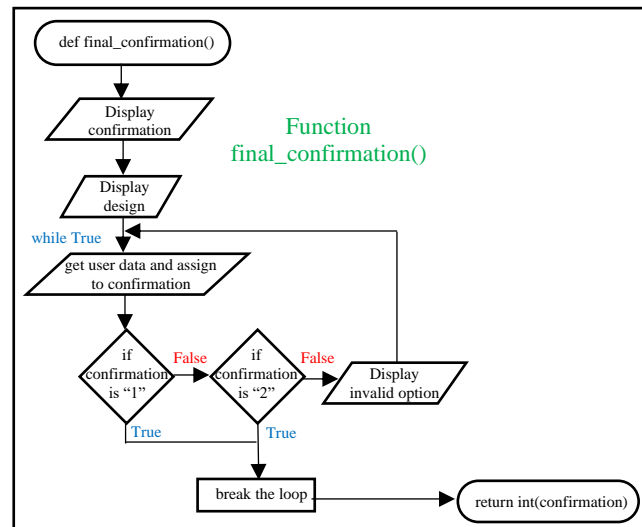
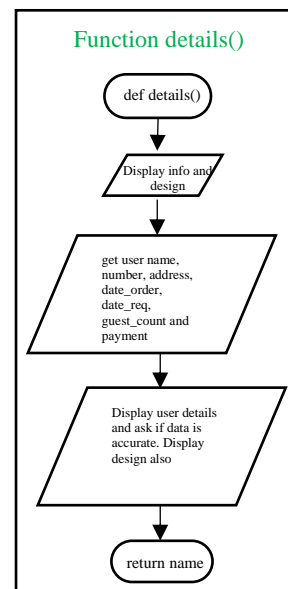
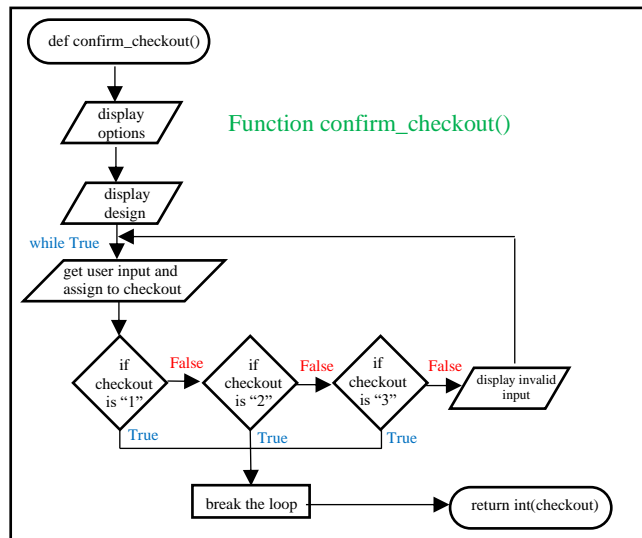


HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

All Cart Conclusion Functions



HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

Definition Of Terms

Variables used in the main loop	
quantity	- used to identify the number of packages/items the user wants to order
ask_resets	- used to identify if user wants to reset his/her order
ask_order_q	- variable used to ask if user want to order or back (return to previous page)
main_option	- option for main options
opt	- option for information options
party_option	- options for party food
party_code	- code of the party pack user wants to order
rental_options	- options for rental and party needs
pack_code	- code of the rental pack user wants to order
item_code	- code of the rental item user wants to order
catering_option	- option for catering services
wed_pack_code	- used to identify the code of the wedding pack user wants to order
bday_pack	- used to identify the code of the birthday pack user wants to order
checkout	- used to identify if user wants to checkout, reset order or back (return to previous page)
name	- used to display the user name
confirmation	- confirmation of user details
order_again_progr am	- identify if user wants to order again or not

Party Pack Variables					
party_pack1	Quantity of Party Packs Ordered in	Party pack 1	price_pp1	Prices of	Party pack 1
party_pack2		Party pack 2	price_pp2		Party pack 2
party_pack3		Party pack 3	price_pp3		Party pack 3
pp1	Display the total of each pack ordered in	Party pack 1	total_party_ pack	Total Costs in Party Pack	
pp2		Party pack 2			
pp3		Party pack 3			

HOLY ANGEL UNIVERSITY

School of Computing 6LOGPROG

Rental Pack Variables					
rental_pack 1	Quantity of Rental Packs Ordered in	Rental pack 1	price_rp1	Prices of	Rental pack 1
rental_pack 2		Rental pack 2	price_rp2		Rental pack 2
rental_pack 3		Rental pack 3	price_rp3		Rental pack 3
item1	Quantity of Rental Items Ordered in	Item 1	price_i1	Prices of	Item 1
item2		Item 2	price_i2		Item 2
item3		Item 3	price_i3		Item 3
item4		Item 4	price_i4		Item 4
item5		Item 5	price_i5		Item 5
item6		Item 6	price_i6		Item 6
p1	Display the total of each pack/item ordered in	Rental pack 1	p6	Display the total of item ordered in	Item 3
p2		Rental pack 2	p7		Item 4
p3		Rental pack 3	p8		Item 5
p4		Item 1	p9		Item 6
p5		Item 2	total_rental s	Total cost in rental and equipment packs/items	

HOLY ANGEL UNIVERSITY

School of Computing 6LOGPROG

Catering Services Variables					
wed_pack1	Quantity of Wedding Packs Ordered in	Wedding Package 1	price_wp1	Prices of	Wedding Package 1
wed_pack2		Wedding Package 2	price_wp2		Wedding Package 2
wed_pack3		Wedding Package 3	price_wp3		Wedding Package 3
b_pack1	Quantity of Rental Items Ordered in	Birthday Package 1	price_bp1	Prices of	Birthday Package 1
b_pack2		Birthday Package 2	price_bp2		Birthday Package 2
b_pack3		Birthday Package 3	price_bp3		Birthday Package 3
b_pack4		Birthday Package 4	price_bp4		Birthday Package 4
b_pack5		Birthday Package 5	price_bp5		Birthday Package 5
b_pack6		Birthday Package 6	price_bp6		Birthday Package 6
c1	Display the total of each pack ordered in	Wedding Package 1	c6	Display the total of each pack ordered in	Birthday Package 3
c2		Wedding Package 2	c7		Birthday Package 4
c3		Wedding Package 3	c8		Birthday Package 5
c4		Birthday Package 1	c9		Birthday Package 6
c5		Birthday Package 2	total_catering_packs	Total cost in catering packs	

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

Main Functions with their respective purpose and variables			
MENU()	- Display the main options of the program and program design		
check_main(x)	- Check if user input is within the given choices and convert it into an integer and return the value. If not, return x.	x	- used to identify user input
four_option(option)	- Check if user input is within the given choices and convert it into an integer and return the value. If not, return option.	option	- used to identify user input
BACK()	- Display design and enter loop. Ask the user to input 0 if he wants to back to the previous page. If user input is not recognized, display invalid input and loop back.	back	- determine user input if he wants to go back to the previous page
six_option()	- Enter loop and ask user for item/pack code with 6 options. If user input is not within the choices, display invalid input and ask again. If user input is within the choices, exit the loop and return user input as an integer.	pack_code	- stores user item/pack code
numberOfItems()	- Enter loop and ask for user input of item/pack quantity. If user input is not an integer, display input must be an integer and ask again. If user input is an integer but less than or equal to 0, display input must be greater than 0 and ask again. Else, break the loop and return the quantity.	quantity	- number of items/packs user wants to order
ask_reset()	- Display choices and enter loop. Ask if user wants to reset order or not. If user input is not within the choices, display invalid input and ask again. Else, break the loop and return user input	ask_reset	- stores user input if he/she wants to reset orders
ask_if_order_q()	- Display choices and enter loop. Ask if user wants order or return to previous page. If user input is not within the choices, display invalid input and ask again. Else, break the loop and return user input	ask_order_q	- stores user input if he/she wants to order or go back to previous page

HOLY ANGEL UNIVERSITY

School of Computing 6LOGPROG

Read More Functions with their respective purpose

information_options()	– Display information options and design
about_us()	– Display BURI catering about us section and design
contact_us()	– Display contacts of BURI catering and design
terms_of_payment()	– Display information about BURI catering terms of payment and design

Party Food Functions with their respective purpose and variables

party_options()	- Display party options and design		
check_party_option(party_option)	- Check if user input is within the given choices and convert it into an integer and return the value. If not, return user input.	party_option	- used if user input is within the choices
PARTY_PACKAGES()	- Display party packages and codes and design		
show_cart_partypacks()	– Display design and check if total of party pack != 0. If it's true, validate conditions, display party pack orders and their total. Else, display no current items in the shopping list.		

Rental and Party Needs Functions with their respective purpose

rental_options()	– Display rental options and design
rentalMenuPacks()	– Display rental menu on packages and design
rentalMenuItems()	– Display rental menu on items and design
show_cart_rentals()	– Display design and check if total of total_rentals != 0. If its true, validate conditions, display rental orders and their total. Else, display no current items in the shopping list.

Catering Services Functions with their respective purpose

catering_options()	– Display catering services options and design
WEDDING_PACKAGES()	– Display wedding packages, codes and design
BIRTHDAY_PACKAGES()	– Display birthday packages, codes and design
show_cart_catering()	– Display design and check if total of total_catering_packs != 0. If its true, validate conditions, display catering packs orders and their total. Else, display no current items in the shopping list.

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

All Cart Conclusion Functions with their respective purpose and variables

confirm_checkout()	– Display checkout options, design and enter loop. Get user input and validate it. If user input is not within the choices, display invalid input and ask again. Else, convert user input to integer and return it.	checkout	- use to get user input if he wants to checkout, reset or return to previous page
details()	– Get user details such as name, number etc., display it back to the user and ask to check it wisely. Finally, program return user input on name	option	- used to identify user input
		name	- get user name
		number	- get user number
		address	- get user venue address
		date_order	- the date which order is created
		date_req	- the date of order delivery
		guest_count	- the number of guests
		payment	- user mode of payment
final_confirmation()	– Display user confirmation of details, design and enter loop. Ask user for confirmation and if user input is not recognized, display invalid option and ask again. Else, break the loop and return user input as an integer	confirmation	– get user input on confirmation of details
final_output()	– Display final outputs, such as name, total bill, how much to pay and all other details.		
order_again()	– Display order again options, design and enter loop. Get user input and validate it. If user input is within the choices, break the loop and return user input as an integer. If not, display invalid option and ask again.	order_again	- get user input about creating another order or not

HOLY ANGEL UNIVERSITY
School of Computing
6LOGPROG

Program Code

```
#-----MAIN FUNCTIONS-----#
def MENU():
    print("O"+"~"*37+****BURI CATERING SERVICE*****+"~"*38+"O")
    print("""\t\t\t[1] Read More
\t\t\t[2] Party Food
\t\t\t[3] Rentals and Party Needs
\t\t\t[4] Catering Services
\t\t\t[5] View your cart
\t\t\t[0] End Program""")
    print("O"+"~"*102+"O")

def check_main(x):
    match x:
        case "1"| "2"| "3"| "4"| "5"| "0":
            x = int(x)
    return x

def BACK():
    print("O"+"~"*102+"O")
    while True:
        back = input("Press [0] to go back: ")
        if back == "0":
            break
        else:
            print("Invalid Input")

def four_option(option):
    match option:
        case "1"| "2"| "3"| "4":
            option = int(option)
    return option

def six_option():
    while True:
        six_option = input("Input Code [1-6]: ")
        match six_option:
            case "1"| "2"| "3"| "4"| "5"| "6":
                break
            case _:
                print("Invalid Input")
    return int(six_option)

def pack_option():
    while True:
        pack_code = input("Input Pack Code [1-3]: ")
        match pack_code:
            case "1"| "2"| "3":
                break
            case _:
                print("Invalid Input")
    return int(pack_code)

def numberOfItems():
    while True:
        try:
            quantity = int(input("Quantity: "))
            if quantity > 0:
                break
            else:
                print("Input must be greater than 0")
        except ValueError:
            print("Input must be an integer")
    return quantity

def ask_reset():
    print("""\t\t\t[R] Reset Order\t\t[B] Back")
    while True:
        ask_reset = input("Input Operation[R/B]: ")
        if ask_reset == "R" or ask_reset == "B":
            break
        else:
            print("Invalid Input")
    return ask_reset

def ask_if_order_q():
    print("""\t\t\t[O] Order\t\t[B] Back")
    while True:
        ask_order_q = input("Input Operation[O/B]: ")
        if ask_order_q == "O" or ask_order_q == "B":
            break
        else:
            print("Invalid Input")
    return ask_order_q

#-----READ MORE FUNCTIONS-----#

def information_options():
    print("O"+"~"*42+****Regarding Us*****+"~"*42+"O")
    print("""\t\t\t[1] About Us
\t\t\t[2] Contact Us
\t\t\t[3] Terms of Payment
\t\t\t[4] Back""")
    print("O"+"~"*102+"O")

def about_us():
    print("O"+"~"*44+****About Us*****+"~"*44+"O")
    print("""n
Since 2018, we at BURI Catering have been serving up joyful occasions. Curiosity led
to the founding of our current catering company, and as we gathered customers and
culinary expertise, we endeavored to enhance our offerings. Carl and Vladimir are the
founders; they know caterers who have studied culinary arts, so they can offer you
with a delectable dinner whenever you need their services. Mark organizes the menus for
```



HOLY ANGEL UNIVERSITY
School of Computing
6LOGPROG

celebrations such as weddings and birthdays. He has catering and eventorganizing experience.

In five years, BURI Catering has established itself as one of the city's caterers (San Fernando, Pampanga) with the most rapid expansion and highest demand. With 30 people that are accessible for weddings and birthdays, we have become one of the most well-known catering businesses. Since the beginning of our careers, we have been recognized as the catering industry's most dependable brand.

The catering business and restaurant continue to expand in popularity, and the restaurant's dedicated team of professionals and personnel give all of its presentations 5-star district-wide appeal. Both large and small catering companies offer social meals and floral and décor arrangements for birthdays and weddings. Remember about BURI Catering.

Call +63123456789 and let us know what we can do for you.\n""")

```
def contact_us():
    print("O"+"="*102+"O")
    print("""\tNatakam ka ba? Hindi mo pa na try? Contact us here!"
```

\tFacebook: Buri Catering
 \tTwitter: @BuriCatering
 \tContact Number: +63917611539
 +63123456789

\Interested at Buri Catering? Please message or look for one of the members below:

\tFounder:
\tVladimir Borja
\tCarl Paldeng

\tCo-founder:
\tMark Erolle Quiambao""")

```
def terms_of_payment():
    print("O" + "="*44 + "Terms of Payment" + "="*42 + "O")
    print("n" + " " * 42, "Catering Agreement(n)")
    print("==== Catering Agreement/General Agreement for Buri.ph. The catering agreement will be
    created once the detail of the client is provided. The client will be required to describe
    the following details: name, date, time, guest count etc. They must also concur with the
    following:
```

\l1. DEPOSIT OF PAYMENT

Fifty percent (50%) of the anticipated cost (COST) is required for order processing. If the order goes through, the deposit amount will be deducted from the final payment.

2. FINAL PAYMENT

The final payment is due three to five days before the event. The payment terms must be adhered to.\n

3. PAYMENT METHOD

All orders are payable with GCASH or CASH. GCASH DETAILS.\n

\t4. GUEST COVERAGE

A final headcount or guest list should be provided at the time of order to avoid further miscalculation.\n

\t5. LEFTOVER

Guests who have leftover foods will be discarded by the Caterer.\n

\t6. BEVERAGES

The guests will be entertained with beverages, or support as agreed upon the agreement.\n

7. STRICTLY NO CANCELLATION OF ORDER

All pre-payments and post-payments won't be refunded if the client cancels the event and notifies the venue within one to three days prior to the event.

\t8. VENUE

The caterer is not responsible for any damage or loss to furnishings, equipment, or clothing from the said venue. However, damages or loss of our products, equipment and more will be the customer's responsibility and must be charged for the damaged items.\n

#-----#

#-----Party Food-----#

```
def party_options():  
    print("O"+"*"**38+"***PARTY PACK OPTIONS***"+"*40+"O")  
    print("*****\t\t\t\t[1]Packs  
\t\t\t\t[2]Cart  
\t\t\t\t[3]Back*****"  
        print("O"+"*"**102+"O"))
```

```
def check_party_option(party_option):
```

```
def check_party_option(party_option):
    match party_option:
        case "1"|"2"|"3":
            party_option = int(party_option)
    return party_option
```

```
def PARTY_PACKAGES():
```

PACKAGE CODE [1]	PACKAGE CODE [2]	PACKAGE CODE [3]
Package 1: Occasional Package (Good for 20 People)	Package 2: Party Package Overload (Good for 20 People)	Package 3: Formal Package (Good for 20 People)
Mixed Veggies and Seafood	Fried Garlic Rice	Roast Beef
Spaghetti & Meatballs	Hotdogs	Fried garlic and butter rice
Plain Rice	Crispy Chicken Tacos	Chicken Cordon Bleu
Garlic Chicken	Pizza Hawaiian	Roast Pork with Oyster Sauce
Roast Beef	Chicken Wings	Pescada en Salsa Verde
Pork and Chicken BBQ	BBQ Pork Sliders	Baked Spaghetti
Fish Fillet	Spaghetti & Meatballs	Mixed Vegetables with Ham
Pork Lumpia	Sandwiches Roasted Beef	Buko Fruit Salad
Iced Tea Bottomless	Iced Tea Bottomless	Iced Tea Bottomless
Ice Mineral Water	Ice Mineral Water	Ice Mineral Water
Price: P 10,261	Price: P 11,707	Price: P 12,404

```
def show_cart_partypacks():
```

```

print(show_cat_partybacks())
print("==41+*****Party Food Cart****")
if total_party_pack != 0:
    print("Package Ordered", "t\t\t\t\t\tQuantity", "t\t\t\t\t\tPrice")
    if party_pack1 != 0:
        print("Party Package 1 [Occasional Package]:", "t\t\t\t\t\t", party_pack1, "t\t\t\t\t\ttP", format(pp1, ',.2f'))
    if party_pack2 != 0:

```



HOLY ANGEL UNIVERSITY
School of Computing
6LOGPROG

```

def main():
    if party_pack2 == 0:
        print("Party Package 2 [Party Package Overload]: ", "\t\t\tP", format(pp2, '.2f'))
        print("Party Package 3 [Formal Package]: ", "\t\t\tP", format(pp3, '.2f'))
        print("\t\t\t**8, Party Pack Total: P", format(total_party_pack, '.2f'))
    else:
        print("\t\t\tThere are no current items in your shopping cart.\n")

party_pack1, party_pack2, party_pack3 = 0, 0, 0 #NUMBER OF PARTY PACKS ORDERED
price_pp1, price_pp2, price_pp3 = 10261.00, 11707.00, 12404.00 #PRICES OF EACH PARTY PACKS
pp1, pp2, pp3 = 0.00, 0.00, 0.00 #DISPLAYS THE TOTAL OF EACH ITEM ORDERED IN EQUIPMENT RENTAL
total_party_pack = 0.00 #TOTAL IN PARTY PACKS

#-----#
#-----RENTALS AND PARTY NEEDS FUNCTIONS-----#
def rental_options():
    print("O "+"+"*33+****RENTAL AND EQUIPMENT OPTIONS*****"+"*35+"O")
    print("""\t\t\t[1] Packs
\t\t\t[2] Individual Items
\t\t\t[3] Cart
\t\t\t[4] Back""")
    print("O "+"+"*102+"O")

def rentalMenuPacks():
    print("O "+"+"*38+****RENTAL PACK OPTIONS*****"+"*39+"O")
    print("""\t\t\t[1] Complete Catering Equipment Rental Package - 100/person
\t\t\t[2] Complete Catering Equipment Rental Package with Waiters - 130/person
\t\t\t[3] Complete Catering Equipment with Flower Arrangement and Waiters - 160/person""")
    print("O "+"+"*102+"O")

def rentalMenuItems():
    print("O "+"+"*38+****RENTAL ITEM OPTIONS*****"+"*39+"O")
    print("""\t\t\t[1] Tents - 100/pc
\t\t\t[2] Tables - 100/pc
\t\t\t[3] Chairs - 20/pc
\t\t\t[4] Table Cloth Linen - 50/pc
\t\t\t[5] Ribbons - 15/pc
\t\t\t[6] Complete Utensils and Silverware - 50/pc""")
    print("O "+"+"*102+"O")

def show_cart_rentals():
    print(" "+"*39+***Equipment Rental Cart**")
    if total_rentals != 0:
        print("Package Ordered", "\t\t\tQuantity", "\t\t\tPrice")
        if rental_pack1 != 0:
            print("Complete Catering Eq. Rental Package:", "\t\t\t", rental_pack1, "\t\t\t\tP", format(p1, '.2f'))
        if rental_pack2 != 0:
            print("Complete Catering Eq. Rental Package w/ Waiters:", "\t\t\t", rental_pack2, "\t\t\t\tP", format(p2, '.2f'))
        if rental_pack3 != 0:
            print("Complete Catering Eq. w/ Flower Arrangement & Waiters:", "\t\t\t", rental_pack3, "\t\t\t\tP", format(p3, '.2f'))
        if item1 != 0:
            print("Tents:", "\t\t\t\t\t", item1, "\t\t\t\t\tP", format(p4, '.2f'))
        if item2 != 0:
            print("Tables:", "\t\t\t\t\t", item2, "\t\t\t\t\tP", format(p5, '.2f'))
        if item3 != 0:
            print("Chairs:", "\t\t\t\t\t", item3, "\t\t\t\t\tP", format(p6, '.2f'))
        if item4 != 0:
            print("Table Cloth Linen:", "\t\t\t\t\t", item4, "\t\t\t\t\tP", format(p7, '.2f'))
        if item5 != 0:
            print("Ribbons:", "\t\t\t\t\t", item5, "\t\t\t\t\tP", format(p8, '.2f'))
        if item6 != 0:
            print("Complete Utensils and Silverware:", "\t\t\t\t\t", item6, "\t\t\t\t\tP", format(p9, '.2f'))
        print("\t\t\t**8, Equipment Total: P", format(total_rentals, '.2f'))
    else:
        print("\t\t\tThere are no current items in your shopping cart.\n")

rental_pack1, rental_pack2, rental_pack3 = 0, 0, 0 #NUMBER OF RENTAL PACKS ORDERED
price_rp1, price_rp2, price_rp3 = 100.00, 130.00, 160.00 #PRICES OF EACH EQUIPMENT PACKS
item1, item2, item3, item4, item5, item6 = 0, 0, 0, 0, 0, 0 #NUMBER OF RENTAL ITEMS ORDERED
price_i1, price_i2, price_i3, price_i4, price_i5, price_i6 = 100.00, 100.00, 20.00, 50.00, 15.00, 50.00 #PRICES OF EACH EQUIPMENT ITEMS
p1, p2, p3, p4, p5, p6, p7, p8, p9 = 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00 #DISPLAYS THE TOTAL OF EACH ITEM ORDERED IN EQUIPMENT RENTAL
total_rentals = 0.00 #TOTAL IN RENTAL AND EQUIPMENT PACKS/ITEMS

#-----#
#-----Catering Services-----#

def catering_options():
    print("O "+"+"*40+****CATERING SERVICES*****"+"*39+"O")
    print("""\t\t\t[1] Wedding Packages
\t\t\t[2] Birthday Packages
\t\t\t[3] Cart
\t\t\t[4] Back""")
    print("O "+"+"*102+"O")

def WEDDING_PACKAGES():
    print("""O=====O
| WEDDING PACKAGE | Set-up and Props Inclusion | Wedding Package |
| 1 Rice Menu | Buffet Set-up | |
| 1 Fish Menu | Formal Uniform for Caterers | Package [1] - P 24,000 |
| 1 Chicken Menu | Complete Wedding Equipments | Good for 30 People |
| 1 Beef Menu | Entrance Arch , Red Carpet, etc. | |
| 1 Vegetable Menu | Invitation Cards & A Special Gift | Package [2] - P38,000 |
| 1 Dessert Menu | Complete Equipment for | Good for 50 People |
| 1 Soup Menu | Catering (Utensils) | |
| Round Bottomless Softdrinks | Complete Catering Equipments | Package [3] - P75,000 |
| Purified Water With Tube Ice | Wedding Party Host and Co-Host | Good for 100 People |
O=====O""")

def BIRTHDAY_PACKAGES():
    print("""O=====O
| BIRTHDAY PACKAGE | Set-up and Props Inclusion | Birthday Packages |
| Plain Rice | Buffet Set-up | |
| Garlic Rice | Formal Uniform for Caterers | Good for N People |
O=====O""")

```



HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

```

| 1 Beef Menu      | Presentable Backdrop with | | | | |
| 1 Chicken Menu   | | Color Motive             | Package [1] - P 18,000 |
| 1 Fish Menu      | | Complete Equipment for   | | 30 People             |
| 1 Vegetable Menu | | Catering (Utensils)       | | |                     |
| 1 Appetizer Dish | | Complete Tables and Chairs | Package [2] - P 28,000 |
| 1 Dessert        | | with Presentable Center Piece | 50 People             |
| 1 Soup           | | and Color Motive         | | |                     |
| Purified Water W/Tube Ice | Presentable Cake Table/Stand | Package [3] - P 55,000 |
| Bottomless Softdrinks | Set-up and Photobooth      | 100 People            |
O=====O
| BIRTHDAY PACKAGE - KIDDIE | Set-up and Props Inclusion | Birthday Packages - Kiddie | | |
| Plain Rice               | Loot Bags and Accent Lights | | |
| 1 Beef Menu              | | Buffet Set-up             | Package [4] - P 13,000 |
| 1 Vegetable Menu         | | Formal Uniform for Caterers | Good for 30 People     |
| Spaghetti                | | Party Props (Hats, Lights, etc.) | | |
| Hotdog                   | | Complete Equipment for     | Package [5] - P 21,000 |
| Cotton Candy Station      | | Catering (Utensils)        | | Good for 50 People    |
| Kiddie Cake              | | Complete Tables and Chairs | | |
| 1 Round Bottomless Softdrinks | with Presentable Center Piece | Package [6] - P 38,000 |
| Purified Water With Tube Ice | and Color Motive           | Good for 100 People    |
O=====O""")

```

```

def show_cart_catering():
    print("\n*42+***Catering Cart***")
    if total_catering_packs != 0:
        print("Package Ordered", "\t\t\t\t\tQuantity", "\t\t\t\t\tPrice")
        if wed_pack1 != 0:
            print("Wedding Package 1:\t\t\t\t\t", wed_pack1, "\t\t\t\t\tP", format(c1, ',.2f'))
        if wed_pack2 != 0:
            print("Wedding Package 2:\t\t\t\t\t", wed_pack2, "\t\t\t\t\tP", format(c2, ',.2f'))
        if wed_pack3 != 0:
            print("Wedding Package 3:\t\t\t\t\t", wed_pack3, "\t\t\t\t\tP", format(c3, ',.2f'))
        if b_pack1 != 0:
            print("Birthday Package 1:\t\t\t\t\t", b_pack1, "\t\t\t\t\tP", format(c4, ',.2f'))
        if b_pack2 != 0:
            print("Birthday Package 2:\t\t\t\t\t", b_pack2, "\t\t\t\t\tP", format(c5, ',.2f'))
        if b_pack3 != 0:
            print("Birthday Package 3:\t\t\t\t\t", b_pack3, "\t\t\t\t\tP", format(c6, ',.2f'))
        if b_pack4 != 0:
            print("Birthday Package 4 - Kiddie:\t\t\t\t\t", b_pack4, "\t\t\t\t\tP", format(c7, ',.2f'))
        if b_pack5 != 0:
            print("Birthday Package 5 - Kiddie:\t\t\t\t\t", b_pack5, "\t\t\t\t\tP", format(c8, ',.2f'))
        if b_pack6 != 0:
            print("Birthday Package 6 - Kiddie:\t\t\t\t\t", b_pack6, "\t\t\t\t\tP", format(c9, ',.2f'))
        print("\n*8,Catering Pack Total: P", format(total_catering_packs, ',.2f'))
    else:
        print("\t\t\t\t\tThere are no current items in your shopping cart.\n")

wed_pack1,wed_pack2,wed_pack3 = 0,0,0 #NUMBER OF WEDDING PACKS ORDERED
price_wp1,price_wp2,price_wp3 = 24000.00, 38000.00, 75000.00 #PRICES OF EACH WEDDING PACKS
b_pack1,b_pack2,b_pack3,b_pack4,b_pack5,b_pack6 = 0,0,0,0,0,0 #NUMBER OF BIRTHDAY PACKS ORDERED
price_bp1,price_bp2,price_bp3,price_bp4,price_bp5,price_bp6 = 18000.00,28000.00,55000.00,13000.00,21000.00,38000.00 #PRICES OF EACH BIRTHDAY PACKS
c1,c2,c3,c4,c5,c6,c7,c8,c9 = 0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00 #DISPLAYS THE TOTAL OF EACH ITEM ORDERED IN CATERING SERVICES
total_catering_packs = 0.00 #TOTAL IN CATERING SERVICES PACKS

#-----#
#-----ALL CART CONCLUSION-----#

def confirm_checkout():
    print("\t\t\t\t\t[1]CHECKOUT\t\t\t\t\t[2]RESET ALL ORDERS\t\t\t\t\t[3]BACK")
    print("\t\t\t\t\t(O+*=*102+O)")
    while True:
        checkout = input("Input Operation[1-3]: ")
        match checkout:
            case "1"|"2"|"3":
                return int(checkout)
            case _:
                print("Invalid Input")

def details():
    print("\t\t\t\t\t(O+*=*102+O)")
    print("Place your details and accurate data. Invalid information will not be processed by the management.")
    name = input("Full Name: ")
    number = input("Contact Number: ")
    address = input("Venue Address: ")
    date_order = input("Input date of order (Ex. January 3,2022): ")
    date_req = input("Input date of order delivery (Ex. January 13, 2022): ")
    guest_count = input("Guest Count: ")
    payment = input("Mode of Payment [GCASH/Paymaya]: ")
    print("\t\t\t\t\t(O+*=*102+O)")
    print("\t\t\t\t\tFull Name:",name,"\t\t\t\t\tContact Number:", number,
          "\t\t\t\t\tVenue Address:",address,"\t\t\t\t\tDate of order:",date_order,
          "\t\t\t\t\tDate of order deadline:",date_req,"\t\t\t\t\tGuest Count:",guest_count,
          "\t\t\t\t\tMode of Payment:",payment,"\t\t\t\t\tTotal Payment P:", format(total_cart, ',.2f'))
    print("\n Make sure to print out all details accurately as our management will check the data submitted.")
    print(" Any form of inaccuracy or invalid data will not be processed. Thank you for your understanding.")
    print("\t\t\t\t\t(O+*=*102+O)")
    return name

def final_confirmation():
    print("\t\t\t\t\tHave you confirmed your details?\t\t\t\t\t[1] Yes\t\t\t\t\t[2] Reset my details")
    print("\t\t\t\t\t(O+*=*102+O)")
    while True:
        confirmation = input("Input Operation: ")
        match confirmation:
            case "1"|"2":
                break
            case _:
                print("Invalid Option")
    return int(confirmation)

def final_output():
    print("\t\t\t\t\t(O+*=*102+O)")
    print("\t\t\t\t\tThank you", name, ""for placing this order. Your order will be processed and you will
    \t\t\t\t\tbe contacted by one of our staff within 1-3 business days.""")

```

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

```

print("\nYour bill came to a total of P", format(total_cart, '.2f'),
      "\nPlease prepare 50% of your total bill which is", format(total_cart/2, '.2f'),
      "\n\n(for the processing of your order.)")
print("\nOnce your order is processed, an agreement will be provided for \n(service finalization.\n
Thank you for being our valued customer! We'll make sure that you will have a blast!\n\n")

def order_again():
    print("O"+"="*102+"O")
    print("\nDo you want to create another order?\nYes[1] No[2] ")
    print("O"+"="*102+"O")
    while True:
        order_again = input("Input Operation: ")
        match order_again:
            case "1"|"2":
                break
            case _:
                print("Invalid Option")
    return int(order_again)

#-----MAIN CODE WHERE THE MAGIC HAPPENS-----#
while True:
    MENU()
    main_opt = input("Input Operation[0-5]: ")
    main_option = check_main(main_opt)

    if main_option == 1:
        while True:
            information_options()
            select = input("Input Operation[1-4]: ")
            opt = four_option(select)
            if opt == 1:
                about_us()
                BACK()
            elif opt == 2:
                contact_us()
                BACK()
            elif opt == 3:
                terms_of_payment()
                BACK()
            elif opt == 4:
                break

            #If opt is 4, break the information options look and go back to MENU()

        else:
            print("Invalid Input")

        #Once done executing if-else, loop back to information_options()

    elif main_option == 2:
        while True:
            party_options()
            answer = input("Input Operation[1-3]: ")
            party_option = check_party_option(answer)

            if party_option == 1:
                PARTY_PACKAGES()
                ask_order_q = ask_if_order_q()
                if ask_order_q == "O":
                    party_code = pack_option()
                    quantity = numberOfItems()
                    if party_code == 1:
                        total_party_pack += price_pp1*quantity
                        party_pack1 += quantity
                        pp1 = party_pack1*price_pp1
                    elif party_code == 2:
                        total_party_pack += price_pp2*quantity
                        party_pack2 += quantity
                        pp2 = party_pack2*price_pp2
                    else:
                        total_party_pack += price_pp3*quantity
                        party_pack3 += quantity
                        pp3 = party_pack3*price_pp3
                print("\nSuccessfully added to your cart!!!\n")

                ## If ask_order_q is B, automatically loop back to party_options()

            elif party_option == 2:
                show_cart_partypacks()
                ask_resets = ask_reset()
                if ask_resets == "R":
                    #RESET PARTY PACKS
                    party_pack1, party_pack2, party_pack3 = 0,0,0

                    pp1, pp2, pp3 = 0.00, 0.00, 0.00

                    total_party_pack = 0.00
                    print("\nSuccessfully Cleared Your Party Cart!!!\n")

                    ## If ask_resets is B, automatically loop back to party_options()

            elif party_option == 3:
                break

            ## If party_option is 3, break the party options loop and go back to MENU()

        else:
            print("Invalid Input")

    elif main_option == 3:
        while True:
            rental_options()
            select = input("Input Operation[1-4]: ")
            rental_option = four_option(select)

```

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

```

if rental_option == 1:
    rentalMenuPacks()
    ask_order_q = ask_if_order_q()
    if ask_order_q == "O":
        pack_code = pack_option()
        quantity = numberOfItems()
        if pack_code == 1:
            total_rentals += price_rp1*quantity
            rental_pack1 += quantity
            p1 = rental_pack1*price_rp1
        elif pack_code == 2:
            total_rentals += price_rp2*quantity
            rental_pack2 += quantity
            p2 = rental_pack2*price_rp2
        else:
            total_rentals += price_rp3*quantity
            rental_pack3 += quantity
            p3 = rental_pack3*price_rp3
        print("\n\t\t\t\t\t[Successfully added to your cart!!!]\n")

        ##Once done ordering, loop back to rental_options()

    ## If ask_order_q is B, automatically loop back to rental_options()

elif rental_option == 2:
    rentalMenuItems()
    ask_order_q = ask_if_order_q()
    if ask_order_q == "O":
        item_code = six_option()
        quantity = numberOfItems()
        if item_code == 1:
            total_rentals += price_i1*quantity
            item1 += quantity
            p4 = item1*price_i1
        elif item_code == 2:
            total_rentals += price_i2*quantity
            item2 += quantity
            p5 = item2*price_i2
        elif item_code == 3:
            total_rentals += price_i3*quantity
            item3 += quantity
            p6 = item3*price_i3
        elif item_code == 4:
            total_rentals += price_i4*quantity
            item4 += quantity
            p7 = item4*price_i4
        elif item_code == 5:
            total_rentals += price_i5*quantity
            item5 += quantity
            p8 = item5*price_i5
        else:
            total_rentals += price_i6*quantity
            item6 += quantity
            p9 = item6*price_i6
        print("\n\t\t\t\t\t[Successfully added to your cart!!!]\n")

        ##Once done ordering, loop back to rental_options()

    ## If ask_order_q is B, automatically loop back to rental_options()

elif rental_option == 3:
    show_cart_rentals()
    ask_resets = ask_reset()
    if ask_resets == "R":
        #RESET RENTAL PACKS

        rental_pack1,rental_pack2,rental_pack3 = 0,0,0
        item1,item2,item3,item4,item5,item6 = 0,0,0,0,0,0
        p1,p2,p3,p4,p5,p6,p7,p8,p9 = 0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00
        total_rentals = 0.00
        print("\n\t\t\t\t\t[Successfully Cleared Your Rental Cart!!!]\n")

        ##Once done resetting, loop back to rental_options()

    ##If ask_reserts is B, automatically loop back to rental_options()

elif rental_option == 4:
    break

    ## If rental_option is 4, break the rental options loop and go back to MENU()

else:
    print("Invalid Input")

elif main_option == 4:
    while True:
        catering_options()
        response = input("Input Operation[1-4]: ")
        catering_option = four_option(response)

        if catering_option == 1:
            WEDDING_PACKAGES()
            ask_order_q = ask_if_order_q()
            if ask_order_q == "O":
                wed_pack_code = pack_option()
                quantity = numberOfItems()
                if wed_pack_code == 1:
                    total_catering_packs += price_wp1*quantity
                    wed_pack1 += quantity
                    c1 = wed_pack1*price_wp1
                elif wed_pack_code == 2:
                    total_catering_packs += price_wp2*quantity
                    wed_pack2 += quantity
                    c2 = wed_pack2*price_wp2
                else:
                    total_catering_packs += price_wp3*quantity
                    wed_pack3 += quantity

```

HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

```

        c3 = wed_pack3*price_wp3
        print("\n\t\t\t[Successfully added to your cart!!!]\n")

        ##Once done ordering, loop back to catering_options()

        ## If ask_order_q is B, automatically loop back to catering_options()

elif catering_option == 2:
    BIRTHDAY_PACKAGES()
    ask_order_q = ask_if_order_q()
    if ask_order_q == "O":
        bday_pack = six_option()
        quantity = numberOfItems()
        if bday_pack == 1:
            total_catering_packs += price_bp1*quantity
            b_pack1 += quantity
            c4 = b_pack1*price_bp1
        elif bday_pack == 2:
            total_catering_packs += price_bp2*quantity
            b_pack2 += quantity
            c5 = b_pack2*price_bp2
        elif bday_pack == 3:
            total_catering_packs += price_bp3*quantity
            b_pack3 += quantity
            c6 = b_pack3*price_bp3
        elif bday_pack == 4:
            total_catering_packs += price_bp4*quantity
            b_pack4 += quantity
            c7 = b_pack4*price_bp4
        elif bday_pack == 5:
            total_catering_packs += price_bp5*quantity
            b_pack5 += quantity
            c8 = b_pack5*price_bp5
        else:
            total_catering_packs += price_bp6*quantity
            b_pack6 += quantity
            c9 = b_pack6*price_bp6
        print("\n\t\t\t[Successfully added to your cart!!!]\n")

        ##Once done ordering, loop back to catering_options()

        ## If ask_order_q is B, automatically loop back to catering_options()

elif catering_option == 3:
    show_cart_catering()
    ask_resets = ask_reset()
    if ask_resets == "R":
        #RESET CATERING PACKS

        wed_pack1,wed_pack2,wed_pack3 = 0,0,0
        b_pack1,b_pack2,b_pack3,b_pack4,b_pack5,b_pack6 = 0,0,0,0,0,0
        c1,c2,c3,c4,c5,c6,c7,c8,c9 = 0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00
        total_catering_packs = 0.00
        print("\n\t\t\t[Successfully Cleared Your Catering Cart!!!]\n")

        ##Once done resetting, loop back to catering_options()

        ##If ask_resets is B, automatically loop back to catering_options()

elif catering_option == 4:
    break

    ## If catering_option is 4, break the catering options loop and go back to MENU()

else:
    print("Invalid Input")

elif main_option == 5:
    if total_party_pack == 0 and total_rentals == 0 and total_catering_packs == 0:
        print("\n\t\t\t[You have no current orders]")
        BACK()
    else:
        total_cart = total_party_pack + total_rentals + total_catering_packs
        print("O"+ "-" * 43 + "***Your Cart***"+ "-" * 44 + "O"+ "\n")
        if total_party_pack != 0:
            show_cart_partypacks()
        if total_rentals != 0:
            show_cart_rentals()
        if total_catering_packs != 0:
            show_cart_catering()
        print("O"+ "-" * 45 + "***Total***"+ "-" * 46 + "O")
        print("\t\t\t*Your current total is P "+str(format(total_cart, '.2f'))
        checkout = confirm_checkout()
        if checkout == 1:
            while True:
                name = details()
                confirmation = final_confirmation()
                if confirmation == 1:
                    final_output()
                    break

            ##If confirmation is 2, automatically loop back to name=details()

order_again_program = order_again()

if order_again_program == 1:
    #RESET PARTY PACKS
    party_pack1,party_pack2,party_pack3 = 0,0,0
    pp1,pp2,pp3 = 0,0,0
    total_party_pack = 0

    #RESET RENTAL PACKS
    rental_pack1,rental_pack2,rental_pack3 = 0,0,0
    item1,item2,item3,item4,item5,item6 = 0,0,0,0,0,0
    p1,p2,p3,p4,p5,p6,p7,p8,p9 = 0,0,0,0,0,0,0,0,0
    total_rentals = 0

```

HOLY ANGEL UNIVERSITY
School of Computing
6LOGPROG

```
#RESET CATERING PACKS
wed_pack1,wed_pack2,wed_pack3 = 0,0,0
b_pack1,b_pack2,b_pack3,b_pack4,b_pack5,b_pack6 = 0,0,0,0,0,0
c1,c2,c3,c4,c5,c6,c7,c8,c9 = 0,0,0,0,0,0,0,0,0
total_catering_packs = 0
else:
    break

#If an order succeeded and the user answers 2 [No], the main loop breaks
elif checkout == 2:
    #RESET PARTY PACKS
    party_pack1,party_pack2,party_pack3 = 0,0,0
    pp1,pp2,pp3 = 0,0,0
    total_party_pack = 0

    #RESET RENTAL PACKS
    rental_pack1,rental_pack2,rental_pack3 = 0,0,0
    item1,item2,item3,item4,item5,item6 = 0,0,0,0,0,0
    p1,p2,p3,p4,p5,p6,p7,p8,p9 = 0,0,0,0,0,0,0,0,0
    total_rentals = 0

    #RESET CATERING PACKS
    wed_pack1,wed_pack2,wed_pack3 = 0,0,0
    b_pack1,b_pack2,b_pack3,b_pack4,b_pack5,b_pack6 = 0,0,0,0,0,0
    c1,c2,c3,c4,c5,c6,c7,c8,c9 = 0,0,0,0,0,0,0,0,0
    total_catering_packs = 0
    print("\n\t\t\t(SUCCESSFULLY CLEARED ALL OF YOUR ORDERS!!!!\n")

##If checkout is 3, automatically loop back to MENU()

elif main_option == 0:
    break

else:
    print("Invalid Input")

print("O"+"-"+"*102"+"O")
print("")
|\t\t\tThanks for using our program.

|\t\t\tPresented by:
|\t\t\tKarl Vladimir Borja
|\t\t\tCarl Lester Paldeng
|\t\t\tMark Erolle Quiambao
|\t\t\tLuis Miguel Cayanan
""")
print("O"+"-"+"*102"+"O")
```

Roles

Member Name	Roles and Contribution
Borja, Karl Vladimir B.	<ul style="list-style-type: none"> - Program Content (Party Packages) - Program Flow - Program testing - Contribute various ideas for program creation
Cayanan, Luis Miguel I.	<ul style="list-style-type: none"> - Program Code - Functions - Input Validation - Menu/Options/Cart/Reset - Program Calculations - Program Flowchart - Program Formatting
Paldeng, Carl Lester P.	<ul style="list-style-type: none"> - Program content (Catering Services) - Created descriptions for Read More (About us, etc.) - Description checker - Program Testing - Contribute ideas on options
Quiambao, Mark Erolle	<ul style="list-style-type: none"> - Aide members in creation of program content such as about us, party packages, etc. - Aide in program formatting - Aide in Program Flow - Program testing and identification of program errors



HOLY ANGEL UNIVERSITY

School of Computing

6LOGPROG

References:

Program References

1. "Jojie's Catering Services Manila." Jojie's Catering Services Accessed October 29, 2022. <https://www.jojies.com/>
2. "Food Ordering System Using Python with Free Source Code Demo." YouTube. YouTube, October 9, 2021. Accessed October 29, 2022. <https://www.youtube.com/watch?v=MZCazGDxUpk&t=248s>
3. hobokengar. "Python Try except ValueError." YouTube. YouTube, September 14, 2013. Accessed October 29, 2022. <https://www.youtube.com/watch?v=Yg7yxLMAb1U>

Party Packages References

1. "Sample Menus · Catering Works." Catering Works, 2022. Accessed October 29, 2022. https://www.cateringworks.com/weddings/wedding_food_and_bev/sample_menus/
2. Egcs. "Birthday Package Diamond." Executive Gourmet, January 29, 2020. Accessed October 29, 2022. <https://www.executivegourmet.ph/birthday-package-diamond/>
3. "Food Catering for Children Birthday Party - Kids Party Catering: Birthday Party Food: Rivera Events." Miami Party Event Planner, April 29, 2022. Accessed October 29, 2022. <https://riveraevents.com/catering-services/kids-buffet/>

Catering Services References

1. "A&J Catering Services." Facebook, 2n.d.. Accessed October 29, 2022. <https://www.facebook.com/jovy.larrios.arthur>
2. "Luxury Events, Valunzuela." Facebook. Accessed October 29, 2022. <https://www.facebook.com/yourluxuryevent>