**Solution of Pre Release Material June 2020**

**Declare PriceForSecSlot = 2**

**Output “WELCOME TO THE MANAGEMENT”**

**// To enter the correct day**

**Output “ Enter 1 for Saturday, 2 for Sunday and 3 for all other days(Monday-Friday)”**

**Input Day**

**// To validate the correct day**

**While (Day<1 OR Day>3) Do**

**Input “Please enter the correct value”, Day**

**EndWhile**

**// DailyTotal is to calculate the total daily earning**

**DailyTotal=0**

**// It is to repeat the loop for next customer**

**CarryOn="yes"**

 **WHILE (CarryOn ="yes" or CarryOn ="YES" or CarryOn ="Y" or CarryOn ="y") Do**

**Output “WELCOME TO CAR PARK”**

**price= 0// It will again become 0 for next customer**

**Input “Do you have a parking number”, Answer**

**If Answer = “Yes” Then**

**Input “Enter the five digit code, Frequent Parking Number”, Number**

**// To validate the frequent parking number, A 5 digit code**

 **If Number>=10000 And Number<=99999 Then**

 **Digit\_5 = Number MOD 10**

 **Number = Number DIV 10**

 **Digit\_4 = Number MOD 10**

 **Number = Number DIV 10**

 **Digit\_3 = Number MOD 10**

 **Number = Number DIV 10**

 **Digit\_2 = Number MOD 10**

 **Digit\_1 = Number DIV 10**

**// To calculate the check digit through modulo 11 method**

 **sum = (1\*Digit\_1)+(2\*Digit\_2)+(3\*Digit\_3)+(4\*Digit\_4)**

 **checkdigit = sum MOD 11**

**// A check to always ensure either the parking number is correct or not (to give discount)**

 **IF checkdigit =digit\_5 Then**

 **PRINT "You got discount"**

 **check="right"**

 **ELSE**

 **PRINT "You did not get discount"**

 **check="wrong"**

 **EndIf**

**Else Print “Your parking number is incorrect” EndIf//End of 5 digit if condition**

**Else Print “OK, No discounts for you” EndIf//End of “Answer = yes” if condition**

**// To input and validate time in hour**

**Input “Please enter the time in hours only, not minutes”, time**

**time= INT(time) // To take only integer part, if a decimal number is entered**

**WHILE (time <8 or time>24) Do**

 **Input "Service is unavailable yet! please try after 8 A.M.", time**

 **time= INT(time)**

 **EndWhile**

**// To input and validate duration in hour**

**Input “Please enter the duration”, hour**

**hour= ROUND(hour) // To round off hour, if a decimal number is entered**

**// To calculate price for Saturday**

 **IF day =1**

// To validate duration for Saturday

 WHILE ((hour>4) DO

 Input "Service cannot be availed more than 4 hours today, Please enter the duration again", hour

 EndWhile

 PricePerHour=3

**// To calculate price for Sunday**

**ElseIF day =2 Then**

**// To validate duration for Sunday**

 **WHILE (hour>8) DO**

 **Input "Service cannot be availed more than 8 hours today, Please enter the duration again", hour**

 **EndWhile**

 **PricePerHour =2**

 **// To calculate price for all working days ( Monday – Friday)**

**ElseIF Day = 3 Then**

**// To validate duration for working days**

 **WHILE (hour>2) DO**

 **Input "Service cannot be availed more than 2 hours today, Please enter the duration again", hour**

 **EndWhile**

**PricePerHour =10**

**EndIf**

 **// The price will be according to the duration, within and outside working hours**

 **FOR i=1 to hour**

 **IF time<16 Then**

 **IF check="right" Then Price =price+( PricePerHour \*0.9)**

 **ELSE Price =price+ PricePerHour EndIF**

 **ELSE:**

 **PriceForSecSlot =2**

 **i= hour // To ensure loop will be terminated when it comes to the second half(Task 3)**

 **IF check="right" Then price=price+( PriceForSecSlot /2)**

 **Else price =price+ PriceForSecSlot EndIf**

 **EndIF**

 **time = time+1// To calculate for next hour**

 **Next i**

**// To check the amount paid (Task 2)**

**INPUT "the payment price is ", price**

**READ PaidPrice**

**WHILE (PaidPrice <price) Do**

 **Input "You must pay equal and more than the shown price", PaidPrice**

**EndWhile**

 **Output "The paid price is ", PaidPrice**

 **DailyTotal = DailyTotal + PaidPrice**

 **Output "Thanks for using our services"**

**// To calculate for next customer**

 **Output "Enter 'yes' to continue and 'no' for the end of the day”, CarryOn**

**EndWhile // To terminate the CarryOn loop**

 **Output "Your daily income is ", DailyTotal**

**Expected Questions**

**Section A**

**1 (a)** All variables, constants and other identifiers must have meaningful names.

Name **two** variables and one constant you could use for **Task 1**. State the data type and purpose of each one.

Variable 1 ......................................................................................................................................

Data type 1 ...............................................................................................................................

Purpose ....................................................................................................................................

...................................................................................................................................................

Variable 2 ......................................................................................................................................

Data type 2 ...............................................................................................................................

Purpose ....................................................................................................................................

...................................................................................................................................................

Constant 1 ......................................................................................................................................

Data type 1 ...............................................................................................................................

Purpose ....................................................................................................................................

...................................................................................................................................................[3]

Reason for use of a Variable .....................................................................................................

...................................................................................................................................................

...................................................................................................................................................[1]

Reason for use of a Constant .....................................................................................................

...................................................................................................................................................

...................................................................................................................................................[1]

**Answer**

Variable 1  **Day**

Data type 1 **Integer**

Purpose **To store day of the week, 1 for Saturday, 2 for Sunday, 3 for Monday-friday**

Variable 2  **CarryOn**

Data type 2 **String**

Purpose **To repeat the loop whenever a new customer comes**

Constant 1  **PriceForSecSlot**

Data type  **Integer**

Purpose **To initialize the price for second slot , which is same for the whole week**

**One** mark for a correct reason, e.g.

Variables allow the storage of values within a program that may change as

the program runs // Variables are used to store values that are input or

calculated

**One** mark for a correct reason, e.g.

Constant is a value that never changes, we use constants in programming because fixed length saves memory.

 **(b)** Describe how you input frequent parking number and verify it with modulo 11 method in **Task 1**.

. ..................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

..................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................[4]

**Answer**

**Input “Enter the five digit code, Frequent Parking Number”, Number**

**// To validate the frequent parking number, A 5 digit code**

 **If Number>=10000 And Number<=99999 Then**

 **Digit\_5 = Number MOD 10**

 **Number = Number DIV 10**

 **Digit\_4 = Number MOD 10**

 **Number = Number DIV 10**

 **Digit\_3 = Number MOD 10**

 **Number = Number DIV 10**

 **Digit\_2 = Number MOD 10**

 **Digit\_1 = Number DIV 10**

**// To calculate the check digit through modulo 11 method**

 **sum = (1\*Digit\_1)+(2\*Digit\_2)+(3\*Digit\_3)+(4\*Digit\_4)**

 **checkdigit = sum MOD 11**

**// A check to always ensure either the parking number is correct or not (to give discount)**

 **IF checkdigit =digit\_5 Then**

 **PRINT "You got discount"**

 **check="right"**

 **ELSE**

 **PRINT "You did not get discount"**

 **check="wrong"**

 **EndIf**

**Else Print “Your parking number is incorrect” EndIf//End of 5 digit if condition**

 **(c)** Write an algorithm for **Task 1**, using **either** pseudocode, programming statements **or** a flowchart. Assume that the first part of **Task 1**, the input of day, time and duration, and input and verification of check digit has already been done.

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................[7]

**Answer**

**IF day =1**

// To validate duration for Saturday

 WHILE ((hour>4) DO

 Input "Service cannot be availed more than 4 hours today, Please enter the duration again", hour

 EndWhile

 PricePerHour=3

**// To calculate price for Sunday**

**ElseIF day =2 Then**

**// To validate duration for Sunday**

 **WHILE (hour>8) DO**

 **Input "Service cannot be availed more than 8 hours today, Please enter the duration again", hour**

 **EndWhile**

 **PricePerHour =2**

 **// To calculate price for all working days ( Monday – Friday)**

**ElseIF Day = 3 Then**

**// To validate duration for working days**

 **WHILE (hour>2) DO**

 **Input "Service cannot be availed more than 2 hours today, Please enter the duration again", hour**

 **EndWhile**

**PricePerHour =10**

**EndIf**

 **// The price will be according to the duration, within and outside working hours**

 **FOR i=1 to hour**

 **IF time<16 Then**

 **IF check="right" Then Price =price+( PricePerHour \*0.9)**

 **ELSE Price =price+ PricePerHour EndIF**

 **ELSE:**

 **PriceForSecSlot =2**

 **i= hour // To ensure loop will be terminated when it comes to the second half(Task 3)**

 **IF check="right" Then price=price+( PriceForSecSlot /2)**

 **Else price =price+ PriceForSecSlot EndIf**

 **EndIF**

 **time = time+1// To calculate for next hour**

 **Next i**

 **(d)** Explain how you could validate that the amount paid by each customer must be greater than or equal to the amount displayed and each customer payment is added to the daily total in **Task 2**. Any programming statements you use in your answer must be fully explained.

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................[3]

**Answer**

**// To check the amount paid (Task 2)**

**When the Price for a customer are calculated, amount paid by him is taken as input, if that is lesser than the Price then it will be taken again by applying a validation check using while do loop, when the amount paid is greater than or equal to the price calculated, it is added in daily total by using a statement of counting.**

**INPUT "the payment price is ", price**

**READ PaidPrice**

**WHILE (PaidPrice <price) Do**

 **Input "You must pay equal and more than the shown price", PaidPrice**

**EndWhile**

 **Output "The paid price is ", PaidPrice**

 **DailyTotal = DailyTotal + PaidPrice**

 **(e)** Explain how you could eliminate the complaint of overcharging in **Task 3**.

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................

...................................................................................................................................................[2]

**Answer**

**When the time becomes equal to or greater than 16, I immediately assign the value of total duration (hour) to loop counter (i) , As “hour” is the upper bound of For loop. As a result the loop is terminated and the amount for Second slot is not added again and again.**