CIE May/June 2016 Pre-release Solution

Computer Science (2210)

Prepared By: Engr. Fahad Khan

In preparation for the examination candidates should attempt the following practical tasks by writing and testing a program(s)
The manager of a building materials delivery service needs a program to check the contents and weight of sacks to ensure that correct orders are made up for delivery. A price for the order will be calculated.
Write and test a program for the manager

- Your program must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly.
- All variables, constans and other identifiers must have meaningful names.

TASK 1-check the contents and weight of a single sack
Each sack must obey the following rules to be accepted:

- Contain cement, gravel or sand, with a letter on the side for easy identification
  - C-cement
  - G-gravel
  - S-sand
- Sand and gravel must weigh over 49.0 and under 50.1 kilograms
- Cement must weigh over 24.9 and under 25.1 kilograms

Input and store the weight and contents for one sack. The contents must be checked and an incorrect sack rejected. The weight must be validated on entry and an overweight or underweight sack rejected.
Output the contents and weight of an accepted sack. If a sack is rejected, output the reason(s).

(Solution of Task 1 using VB.NET Programming)

Note: The lines written in green colour such as this one:

```
'These green lines starting with apostrophe are for explanation purpose only and no need
to remember them. You are only supposed to remember and understand the programming code.
```

```
Module Module1
    'Main function is entry point to any program
    Sub Main()

        'Declare content variable with char (character) data type to store content of
sack
        Dim content As Char
        'Declare weight variable to store weight of sack
        Dim weight As Single

        'Console.WriteLine function is used to show output message or values
        Console.WriteLine("Enter the content of a sack, C for cement sack, G for gravel
sack and S for sand sack")
```

```
        'Console.ReadLine is used to take input value and store it in variable on left
side of euqal sign
        content = Console.ReadLine

        'Using IF statement to check whether the character entered is C which is Cement
        If (content = "C") Then
            'Using Do Loop Until to accept only valid values
            'Do Loop Until will keep on repeating until input value of weight is in
between 24.9 and 25.1
            Do
                Console.WriteLine("Enter weight of cement sack in between 24.9KG and
25.1KG")
                'Taking value of weight as input using Console.ReadLine function and
storing it in weight variable
                weight = Console.ReadLine

                'Checking whether weight is less than or equals to 24.9 then it is
underweight
                If (weight <= 24.9) Then
                    Console.WriteLine("Cement sack is underweight")
                    'If statement is always ended with End If keyword
                End If

                'Checking whether weight is greater than or equals to 25.1 then it is
overweight
                If (weight >= 25.1) Then
                    Console.WriteLine("Cement sack is overweight")
                End If
                'Do Loop Until condition is specified here
                'Loop should repeat until weight value is in between 24.9 And 25.1
            Loop Until (weight > 24.9 And weight < 25.1)

            'Displaying content of the sack in output
            Console.WriteLine("The content of sack is {0}", content)
            'Displaying the weight of sack in output
            Console.WriteLine("The weight of sack is {0}KG", weight)

            'If content is equal to C then this ElseIF part for G or S will not execute
            'If content is not equal to C then this ElseIf part for G or S will execute
        ElseIf (content = "G" Or content = "S") Then

            'Using Do Loop Until will keep on repeating until input value of weight is in
between 49.0 and 50.1
            Do
                'Displaying message to enter weight of gravel or sand sack depending what
is content
                Console.WriteLine("Enter weight of {0} sack in between 49.0KG and
50.1KG", content)
                'Taking weight value as input
                weight = Console.ReadLine
                'Checking whether weight is less than or equals to 49.0 then it is
underweight
                If (weight <= 49.0) Then
                    Console.WriteLine("{0} sack is underweight", content)
                End If
                'Checking whether weight is greater than or equals to 50.1 then it is
overweight
                If (weight >= 50.1) Then
```

```
                    Console.WriteLine("{0} sack is overweight", content)
                End If
                'Do Loop Until condition is specified here
                'Loop should repeat until weight value is in between 49.0 And 50.1
            Loop Until (weight > 49.0 And weight < 50.1)

            'Displaying content of sack in output
            Console.WriteLine("The content of sack is {0}", content)
            'Displaying weight of sack in output
            Console.WriteLine("The weight of sack is {0}KG", weight)

            'If content is not C, G or S then Else part will execute
        Else
            'Displaying message that entered content is incorrect
            Console.WriteLine("The entered content is incorrect")
            'Ending very first IF statement here with End If
        End If

        'Console.ReadKey function is used to hold output on screen until a button on
keyboard is pressed
        Console.ReadKey()
        'Main function is ended here with End Sub keyword
    End Sub
    'Module1 is ended here with End Module keyword
End Module
```

(Solution of Task 1 using Pseudocode)

```
DECLARE content     : CHAR
DECLARE weight      : REAL
WRITE "Enter the content of a sack, C for cement, G for gravel and S for sand sack"
READ content
IF content = 'C'
THEN
REPEAT
WRITE "Enter weight of cement sack in between 24.9KG and 25.1KG"
READ weight
IF weight < = 24.9 THEN WRITE "Sack is underweight"
IF weight > = 25.1   THEN WRITE "Sack is overweight"
UNTIL weight > 24.9 and weight < 25.1
WRITE "The content of sack is:" , content
WRITE "The weight of sack is:" , weight
```

```
ELSEIF content = ‘G’ or content = ‘S’
THEN
REPEAT
WRITE “Enter weight of sack between 49.0KG and 50.1KG”
READ weight
IF weight < = 49.0 THEN WRITE “Sack is underweight”
IF weight > = 50.1   THEN WRITE “Sack is overweight”
UNTIL weight > 49.0 and weight < 50.1
WRITE “The content of sack is:” , content
WRITE “The weight of sack is:” , weight
ELSE
WRITE “The Content is Invalid”
ENDIF
```

TASK 2- check a customer’s order for delivery
Input and store the number of sacks of each type required for the order. Use TASK 1 to check the contents and weigh of each sack. Ensure that the delivery contains the correct number and type of sacks for the order.
Output the total weight of the order.
Output the number of sacks rejected from the order.

(Solution of Task 1 & Task 2 using VB.NET Programming)

Note: Task2 is very simple and you need to add few lines in task1 to reach task 2. New added lines of task 2 will be bold while lines of task 1 will remain normal.

```
Module Module1
    'Main function is entry point to any program
    Sub Main()

        'Declare content variable with char (character) data type to store content of
sack
        Dim content As Char
        'Declare weight variable to store weight of sack
        Dim weight As Single

         'Declare variable to store number of cement sacks
         Dim c_sack As Integer
         'Declare variable to store number of gravel sacks
         Dim g_sack As Integer
        'Declare variable to store number of sand sacks
        Dim s_sack As Integer
        'Declare variable to store total order of cement, gravel and sand sacks
        Dim total_order As Integer

        'Declare total_weight variable to store total weight of all sacks
        Dim total_weight As Single
        'Declare rej variable to count number of sacks rejected
        Dim rej As Integer
        'Declare For Loop count variable here
        Dim count As Integer

       'Assign 0 values to both rej and total_weight variable
        rej = 0
        total_weight = 0


        Console.WriteLine("Enter the number of cement sacks")
        'Storing number of cement sacks in c_sack variable
        c_sack = Console.ReadLine()
        Console.WriteLine("Enter the number of gravel sacks")
        'Storing number of gravel sacks in g_sack variable
        g_sack = Console.ReadLine()
        Console.WriteLine("Enter the number of sand sacks")
        'Storing number of sand sacks in s_sack variable
        s_sack = Console.ReadLine()

       'calculating value of total order
        total_order = c_sack + g_sack + s_sack

        'Using FOR loop which will repeat and take content & weight values for
total order
        For count = 1 To total_order

            'Console.WriteLine function is used to show output message or values
            Console.WriteLine("Enter the content of a sack, C for cement sack, G for
gravel sack and S for sand sack")
            'Console.ReadLine is used to take input value and store it in variable on
left side of euqal sign
            content = Console.ReadLine

        'Using IF statement to check whether the character entered is C which is Cement
            If (content = "C") Then
                'Using Do Loop Until to accept only valid values
                'Do Loop Until will keep on repeating until input value of weight is in
between 24.9 and 25.1
                Do
                    Console.WriteLine("Enter weight of cement sack in between 24.9KG and
25.1KG")
                    'Taking value of weight as input using Console.ReadLine function and
storing it in weight variable
                    weight = Console.ReadLine
```

```
      'Checking whether weight is less than or equals to 24.9 then it is underweight
          If (weight <= 24.9) Then
            Console.WriteLine("Cement sack is underweight")
         'Underweight sack is rejected so rej variable is incremented by 1
             rej = rej + 1
            'If statement is always ended with End If keyword
          End If

     'Checking whether weight is greater than or equals to 25.1 then it is overweight
          If (weight >= 25.1) Then
            Console.WriteLine("Cement sack is overweight")
          'Overweight sack is rejected so rej variable is incremented by 1
             rej = rej + 1
          End If
          'Do Loop Until condition is specified here
          'Loop should repeat until weight value is in between 24.9 And 25.1
        Loop Until (weight > 24.9 And weight < 25.1)

        'Displaying content of the sack in output
        Console.WriteLine("The content of sack is {0}", content)
        'Displaying the weight of sack in output
        Console.WriteLine("The weight of sack is {0}KG", weight)

       'If content is equal to C then this ElseIF part for G or S will not execute
       'If content is not equal to C then this ElseIf part for G or S will execute
      ElseIf (content = "G" Or content = "S") Then

        'Using Do Loop Until will keep on repeating until input value of weight
is in between 49.0 and 50.1
        Do
   'Displaying message to enter weight of gravel or sand sack depending what is content
          Console.WriteLine("Enter weight of {0} sack in between 49.0KG and
50.1KG", content)
          'Taking weight value as input
          weight = Console.ReadLine
     'Checking whether weight is less than or equals to 49.0 then it is underweight
          If (weight <= 49.0) Then
            Console.WriteLine("{0} sack is underweight", content)
         'Underweight sack is rejected so rej variable is incremented by 1
             rej = rej + 1
          End If
     'Checking whether weight is greater than or equals to 50.1 then it is overweight
          If (weight >= 50.1) Then
            Console.WriteLine("{0} sack is overweight", content)
         'Overweight sack is rejected so rej variable is incremented by 1
             rej = rej + 1
          End If
          'Do Loop Until condition is specified here
          'Loop should repeat until weight value is in between 49.0 And 50.1
        Loop Until (weight > 49.0 And weight < 50.1)

        'Displaying content of sack in output
        Console.WriteLine("The content of sack is {0}", content)
        'Displaying weight of sack in output
        Console.WriteLine("The weight of sack is {0}KG", weight)
```

```
                'If content is not C, G or S then Else part will execute
            Else
                'Displaying message that entered content is incorrect
                Console.WriteLine("The entered content is incorrect")
                'Ending very first IF statement here with End If
            End If
            'Calculating total weight here
             total_weight = total_weight + weight
        Next
         'Blank Console.WriteLine function is used to give one line space
         Console.WriteLine()
         'Displaying total weight in output
         Console.WriteLine("The total weight of order is: {0}", total_weight)
         'Displaying number of rejected sacks in output
         Console.WriteLine("The number of sacks rejected are: {0}", rej)

        'Console.ReadKey function is used to hold output on screen until a button on
keyboard is pressed
        Console.ReadKey()
        'Main function is ended here with End Sub keyword
    End Sub
    'Module1 is ended here with End Module keyword
End Module
```

(Solution of Task 1 & 2 using Pseudocode)

```
DECLARE content                                   : CHAR
DECLARE weight, total_weight                      : REAL
DECLARE c_sack, s_sack, g_sack, total_order       :INTEGER
DECLARE rej, count                                :INTEGER
rej = 0
total_weight = 0
WRITE "Enter number of cement, gravel and sand sacks"
READ c_sack
READ g_sack
READ s_sack
total_order = c_sack + g_sack + s_sack
FOR count ← 1 TO total_order
WRITE "Enter the content of a sack, C for cement, G for gravel and S for sand sack"
READ content
IF content = 'C'
THEN
REPEAT
WRITE "Enter weight of cement sack in between 24.9KG and 25.1KG"
READ weight
```

```
IF weight < = 24.9
THEN
WRITE “Sack is underweight”
rej ← rej + 1
ENDIF
IF weight > = 25.1
THEN
WRITE “Sack is overweight”
rej ← rej + 1
UNTIL weight > 24.9 and weight < 25.1
WRITE “The content of sack is:” , content
WRITE “The weight of sack is:” , weight
ELSEIF content = ‘G’ or content = ‘S’
THEN
REPEAT
WRITE “Enter weight of sack between 49.0KG and 50.1KG”
READ weight
IF weight < = 49.0
THEN
WRITE “Sack is underweight”
rej ← rej + 1
ENDIF
IF weight > = 50.1
THEN
WRITE “Sack is overweight”
rej ← rej + 1
ENDIF
UNTIL weight > 49.0 and weight < 50.1
WRITE “The content of sack is:” , content
WRITE “The weight of sack is:” , weight
ELSE
WRITE “The Content is Invalid”
ENDIF
total_weight ← total_weight + weight
NEXT
WRITE “Total weight of order is:” , total_weight
WRITE “Number of rejected sacks are:” , rej
```

TASK 3- calculate the price for a customer’s order

Extend TASK 2 to calculate a price for an order. Prices for the sacks are as follows:

- Regular price for each sack
    - Cement,$3
    - Gravel,$2
    - Sand,$2
- Discount price for a special pack containing 1 sack of cement,2 sacks of sand and 2 sacks of gravel,$10

Calculate and output the regular price for the order. Check how many special packs are in the order. If a discount price applies then output the new price for the order and the amount saved.

(Solution of Task 3 using VB.NET Programming)

```
       'Declare variable to store the number of special packs
        Dim sp As Integer

        'Declare the variable to store the value of special pack price
        Dim sp_price As Integer

        'Declare the variable to store actual price of order
        Dim actual_price As Integer

        'Declare the variable to store discount price
        Dim discount_price As Integer

        'Declare variable to store the total discount
        Dim total_discount As Integer

'Variables c_sack, g_sack and s_sack are already declared in task 1 with integer data
type. Values for all these three variables are also taken as input in Task 1.

'Calculating actual price of order
actual_price = (c_sack * 3) + (g_sack * 2) + (s_sack * 2)

Do
            'Checking if condition for special pack
            If c_sack >= 1 And g_sack >= 2 And s_sack >= 2 Then
                'If condition is true then sp variable is incremented by 1
                'sp variable tells total number of special packs in order
                sp = sp + 1
         '1 is subtracted from c_sack because 1 cement sack is taken out for special pack
                c_sack = c_sack - 1
       '2 is subtracted from g_sack because 2 gravel sacks are taken out for special pack
                g_sack = g_sack - 2
       '2 is subtracted from s_sack because 2 sand sacks are taken out for special pack
                s_sack = s_sack - 2
            Else
                'If condition for special pack is not true then Else part will execute
           Console.WriteLine("Oder is not a special pack, instead it is a regular order")
                'IF statement is ending here
            End If
```

```
'Loop will continue to calculate the special packs
'Loop will stop when there is zero cement sack or 1 gravel sack or 1 sand sack are left
        Loop Until c_sack = 0 Or g_sack = 1 Or s_sack = 1

'This condition will be true If the number of special packs (sp) are greater than or
equals to 1
        If sp >= 1 Then

'Multiplying number of special packs with price of special pack which is 10 dollars
            sp_price = sp * 10
            'Displaying total special packs in output
            Console.WriteLine("Total special packs are: {0}", sp)
            'Displaying price of special packs in output
            Console.WriteLine("Price of special packs in dollars is: {0}", sp_price)

'Caclulating discount price by adding price of special packs and price of all remaining
sacks
            discount_price = (c_sack * 3) + (g_sack * 2) + (s_sack * 2) + sp_price

           'Displaying actual price of order in output
            Console.WriteLine("The actual price of order is: {0} ", actual_price)
            'Displaying discounted price of order in output
            Console.WriteLine("The discounted price of order is: {0}", discount_price)
            'Calculating total discount on the order which includes special packs
            total_discount = actual_price - discount_price
            'Displaying total discount on the order in output
            Console.WriteLine("Total discount in order is: {0}", total_discount)

'If there is no special pack in order then it is regular order and Else part will execute
        Else

'Displaying the regular or actual price of order in output
  Console.WriteLine("Price of regular order in dollars is: {0}", actual_price)
 'IF statement is ending here at End If
        End If
```

(Solution of Task 3 using Pseudocode)

```
DECLARE  sp, sp_price, , actual price                    :INTEGER
DECLARE discount_price, total_discount                   :INTEGER
actual_price ← (c_sack * 3) + (g_sack * 2) + (s_sack * 2)
REPEAT
IF c_sack >= 1 And g_sack >= 2 And s_sack >= 2
THEN
sp ← sp + 1
c_sack ← c_sack - 1
g_sack ← g_sack -2
s_sack ← s_sack - 2
ELSE
WRITE “Order is not a special pack”
ENDIF
UNTIL c_sack = 0  or  g_sack = 0  or  s_sack = 0
```

```
IF sp >= 1
THEN
sp_price ← sp_price * 10
WRITE “Total special packs are” , sp
WRITE “Total price of special packs in dollars”, sp_price
discount_price ← (c_sack * 3) + (g_sack * 2) + (s_sack * 2) + sp_price
WRITE “ The actual price of order is”, actual_price
WRITE “The discounted price of order is” , discount_price
total_discount ← actual_price - discount_price
WRITE “Total discount in dollars for the order is” , total_discount
ELSE
WRITE “Price of regular order is” , actual_price
ENDIF
```

(Solution of whole Pre-release using VB.NET Programming with Explanation)

```
Module Module1

    Sub Main()

'Declare variables to store the number of cement sacks (c_sack) , gravel sacks and sand sacks.
        Dim c_sack As Integer
        Dim g_sack As Integer
        Dim s_sack As Integer

        'Declare variable to store the number of special packs
        Dim sp As Integer

        'Declare the variable to store the value of special pack price
        Dim sp_price As Integer

        'Declare the variable to store actual price of order
        Dim actual_price As Integer

        'Declare the variable to store discount price
        Dim discount_price As Integer

        'Declare variable to store the total discount
        Dim total_discount As Integer

        'Declare variable to store the content of a sack e-g cement, sand or gravel.
        Dim content As Char

        'Declare variable to store weight of sack
        Dim weight As Single

        'Declare variable to store the number of rejected sacks
        Dim rej As Integer
        'Declare variable to store the total weight of all sacks
        Dim total_weight As Single

        'Declare total order to store the total number of sacks in an order
        Dim total_order As Integer

        'Assigning 0 value to both rej and total_weight variables
        rej = 0
        total_weight = 0

        'Console.WriteLine function is used to output a message
        'Console.ReadLine function is used to get input
        'Displaying message to enter the number of cement sacks
        Console.WriteLine("Enter the number of cement sacks")

        'Number of cement sacks are taken as input and stored in c_sack variable
        c_sack = Console.ReadLine()

        'Displaying message to enter the number of gravel sacks
        Console.WriteLine("Enter the number of gravel sacks")

        'Number of gravel sacks are taken as input and stored in g_sack variable
        g_sack = Console.ReadLine()

        'Displaying message to enter the number of sand sacks
        Console.WriteLine("Enter the number of sand sacks")

        'Number of gravel sacks are taken as input and stored in s_sack variable
        s_sack = Console.ReadLine()

        'Because price of a cement sack is $3, gravel sack is $2 and sand sack is $2.
        'All these prices are given in task 3 of pre-release material
        'Multiplying number of sacks of each type with their price to calculate total or
actual price
        actual_price = (c_sack * 3) + (g_sack * 2) + (s_sack * 2)

        'Adding all type of sacks to caclulate total sacks in order and storing it in
total_order variable
        total_order = c_sack + g_sack + s_sack

        'For loop is used to repeat programming code for pre-determined number of times
        'Here for loop will repeat to take input values for all sack and total number of
sacks are stored in total_order
        'So For loop will keep on repeating until its count is greater than the
total_order
        For count As Integer = 1 To total_order

            Console.WriteLine()
       'Take the content of sack as input either C, G or S and store in content variable
Console.WriteLine("Enter the content of a sack, C for cement, G for gravel and S for
sand")
            content = Console.ReadLine
            'Use IF statement to check if content is C or not
            'IF content is C then following statements will execute
            If (content = "C") Then

                'Do Loop UNTIL is used as check point.
                'It wil only pass cement weight value between 24.9KG and 25.1KG
```

```
           Do

      Console.WriteLine("Enter weight of cement sack in between 24.9KG and 25.1KG")
               weight = Console.ReadLine

               'Using the IF statement to check whether cement sack is underweight
               If (weight <= 24.9) Then
                   Console.WriteLine("Cement sack is underweight")
             'If it is underweight then rejected so rej variable is incremented by 1
                   rej = rej + 1
               End If

               'Using the IF statement to check whether cement sack is overweight
               If (weight >= 25.1) Then
                   Console.WriteLine("Cement sack is overweight")
              'If it is overweight then rejected so rej variable is incremented by 1
                   rej = rej + 1
               End If

               'Condition of DO Loop Until to check valid weight values
           Loop Until (weight > 24.9 And weight < 25.1)

           'Displaying the content of sack
           Console.WriteLine("The content of sack is {0}", content)
           'Displaying the weight of sack
           Console.WriteLine("The weight of sack is {0}KG", weight)


       'IF content is not C and it is G or S then following statements will execute
       ElseIf (content = "G" Or content = "S") Then

           'Do Loop UNTIL is used as check point.
           'It wil only pass Gravel or Sack weight value between 49.0KG and 50.1KG
           Do

Console.WriteLine("Enter weight of {0} sack in between 49.0KG and 50.1KG", content)
               weight = Console.ReadLine

        'Using the IF statement to check whether Gravel or Sand sack is underweight
               If (weight <= 49.0) Then
                   Console.WriteLine("{0} sack is underweight", content)
             'If it is underweight then rejected so rej variable is incremented by 1
                   rej = rej + 1
               End If

         'Using the IF statement to check whether Gravel or Sand sack is Overweight
               If (weight >= 50.1) Then
                   Console.WriteLine("{0} sack is overweight", content)
             'If it is Overweight then rejected so rej variable is incremented by 1
                   rej = rej + 1
               End If
               'Condition of DO Loop Until to check valid weight values
           Loop Until (weight > 49.0 And weight < 50.1)

           'Displaying the content of sack
           Console.WriteLine("The content of sack is {0}", content)
           'Displaying the weight of sack
           Console.WriteLine("The weight of sack is {0}KG", weight)
```

```
                'If the content is not C, G or S then Else statement will be executed
            Else
                Console.WriteLine("The entered content is incorrect")

                'End If is indicating the end of IF statement
            End If

            'Caculating total weight of sacks here
            total_weight = total_weight + weight

            'For loop is ending here
        Next

        'The blank Console.WriteLine function is used for one line space
        Console.WriteLine()

        'Displaying total weight of order
        Console.WriteLine("The total weight of order is: {0}", total_weight)
        'Displaying the number of rejected sacks
        Console.WriteLine("The number of sacks rejected are: {0}", rej)

        'Till this point task 1 and task 2 are solved

        'Now solving task 3 of pre-release scenario

   'We will use Do Loop Until for calculating Special Packs
   'As per task 3, a special pack contains 1 cement sack, 2 gravel sacks and 2 sand sacks
        Do
            'Checking if condition for special pack
            If c_sack >= 1 And g_sack >= 2 And s_sack >= 2 Then
                'If condition is true then sp variable is incremented by 1
                'sp variable tells total number of special packs in order
                sp = sp + 1
       '1 is subtracted from c_sack because 1 cement sack is taken out for special pack
                c_sack = c_sack - 1
       '2 is subtracted from g_sack because 2 gravel sacks are taken out for special pack
                g_sack = g_sack - 2
        '2 is subtracted from s_sack because 2 sand sacks are taken out for special pack
                s_sack = s_sack - 2
            Else
                'If condition for special pack is not true then Else part will execute
           Console.WriteLine("Oder is not a special pack, instead it is a regular order")
                'IF statement is ending here
            End If
  'Loop will continue to calculate the special packs
  'Loop will stop when there is zero cement sack or 1 gravel sack or 1 sand sack are left
        Loop Until c_sack = 0 Or g_sack = 1 Or s_sack = 1

  'This condition will be true If the number of special packs (sp) are greater than or
equals to 1
        If sp >= 1 Then
      'Multiplying number of special packs with price of special pack which is 10 dollars
            sp_price = sp * 10
            'Displaying total special packs in output
            Console.WriteLine("Total special packs are: {0}", sp)
            'Displaying price of special packs in output
            Console.WriteLine("Price of special packs in dollars is: {0}", sp_price)
```

```
'Caclulating discount price by adding price of special packs and price of all remaining
sacks
            discount_price = (c_sack * 3) + (g_sack * 2) + (s_sack * 2) + sp_price
            'Displaying actual price of order in output
            Console.WriteLine("The actual price of order is: {0} ", actual_price)
            'Displaying discounted price of order in output
            Console.WriteLine("The discounted price of order is: {0}", discount_price)
            'Calculating total discount on the order which includes special packs
            total_discount = actual_price - discount_price
            'Displaying total discount on the order in output
            Console.WriteLine("Total discount in order is: {0}", total_discount)

        Else
'If there is no special pack in order then it is regular order and Else part will execute
'The regular price of order is caclulated by multiplying number of sacks with their
prices and then adding these prices
            regular_order_price = (c_sack * 3) + (g_sack * 2) + (s_sack * 2)
            'Displaying the regular price in output
    Console.WriteLine("Price of regular order in dollars is: {0}", regular_order_price)
            'IF statement is ending here at End If
        End If

        'Console.ReadKey is used to hold output on screen until any key is pressed
        Console.ReadKey()

        'End Sub is indicating the end of Main method
    End Sub

End Module
```

(Solution of whole Pre-release using VB.NET Programming without Explanation)

```
Module Module1
Sub Main()
Dim c_sack, g_sack, s_sack, total_order, sp, sp_price As Integer
Dim actual_price, discount_price, total_discount, rej As Integer
Dim content As Char
Dim weight, total_weight As Single

rej = 0
total_weight = 0

Console.WriteLine("Enter the number of cement sacks")
c_sack = Console.ReadLine()
Console.WriteLine("Enter the number of gravel sacks")
g_sack = Console.ReadLine()
Console.WriteLine("Enter the number of sand sacks")
```

```
s_sack = Console.ReadLine()

actual_price = (c_sack * 3) + (g_sack * 2) + (s_sack * 2)
total_order = c_sack + g_sack + s_sack

For count As Integer = 1 To total_order
Console.WriteLine()
Console.WriteLine("Enter the content of a sack, C for cement, G for gravel and S for sand")
content = Console.ReadLine

If (content = "C") Then
Do
Console.WriteLine("Enter weight of cement sack in between 24.9KG and 25.1KG")
weight = Console.ReadLine

If (weight <= 24.9) Then
Console.WriteLine("Cement sack is underweight")
rej = rej + 1
End If

If (weight >= 25.1) Then
Console.WriteLine("Cement sack is overweight")
rej = rej + 1
End If
Loop Until (weight > 24.9 And weight < 25.1)
Console.WriteLine("The content of sack is {0}", content)
Console.WriteLine("The weight of sack is {0}KG", weight)

ElseIf (content = "G" Or content = "S") Then
Do
Console.WriteLine("Enter weight of {0} sack in between 49.0KG and 50.1KG", content)
weight = Console.ReadLine

If (weight <= 49.0) Then
Console.WriteLine("{0} sack is underweight", content)
rej = rej + 1
End If

If (weight >= 50.1) Then
Console.WriteLine("{0} sack is overweight", content)
```

```
rej = rej + 1
End If
Loop Until (weight > 49.0 And weight < 50.1)

Console.WriteLine("The content of sack is {0}", content)
Console.WriteLine("The weight of sack is {0}KG", weight)

Else
Console.WriteLine("The entered content is incorrect")
End If

total_weight = total_weight + weight
Next

Console.WriteLine()
Console.WriteLine("The total weight of order is: {0}", total_weight)
Console.WriteLine("The number of sacks rejected are: {0}", rej)

Do
If c_sack >= 1 And g_sack >= 2 And s_sack >= 2 Then
sp = sp + 1
c_sack = c_sack - 1
g_sack = g_sack - 2
s_sack = s_sack - 2
Else
Console.WriteLine("Oder is not a special pack, instead it is a regular order")
End If
Loop Until c_sack = 0 Or g_sack = 1 Or s_sack = 1

If sp >= 1 Then
sp_price = sp * 10
Console.WriteLine("Total special packs are: {0}", sp)
Console.WriteLine("Price of special packs in dollars is: {0}", sp_price)
discount_price = (c_sack * 3) + (g_sack * 2) + (s_sack * 2) + sp_price
Console.WriteLine("The actual price of order is: {0} ", actual_price)
Console.WriteLine("The discounted price of order is: {0}", discount_price)
total_discount = actual_price - discount_price
Console.WriteLine("Total discount in order is: {0}", total_discount)
```

```
Else
Console.WriteLine("Price of regular order in dollars is: {0}", actual_price)
End If

Console.ReadKey()
End Sub
End Module
```

(Solution of whole Pre-release using Pseudocode)

```
DECLARE content                                                   : CHAR
DECLARE weight, total_weight                                      : REAL
DECLARE c_sack, s_sack, g_sack, total_order, sp, sp_price         :INTEGER
DECLARE rej, count, actual price, discount_price, total_discount  :INTEGER

rej = 0
total_weight = 0
WRITE “Enter number of cement, gravel and sand sacks”
READ c_sack
READ g_sack
READ s_sack
total_order = c_sack + g_sack + s_sack
FOR count ← 1 TO total_order
WRITE “Enter the content of a sack, C for cement, G for gravel and S for sand sack”
READ content
IF content = ‘C’
THEN
REPEAT
WRITE “Enter weight of cement sack in between 24.9KG and 25.1KG”
READ weight

IF weight < = 24.9
THEN
WRITE “Sack is underweight”
rej ← rej + 1
ENDIF
IF weight > = 25.1
THEN
WRITE “Sack is overweight”
```

```
rej ← rej + 1
UNTIL weight > 24.9 and weight < 25.1
WRITE “The content of sack is:” , content
WRITE “The weight of sack is:” , weight
ELSEIF content = ‘G’ or content = ‘S’
THEN
REPEAT
WRITE “Enter weight of sack between 49.0KG and 50.1KG”
READ weight
IF weight < = 49.0
THEN
WRITE “Sack is underweight”
rej ← rej + 1
ENDIF
IF weight > = 50.1
THEN
WRITE “Sack is overweight”
rej ← rej + 1
ENDIF
UNTIL weight > 49.0 and weight < 50.1
WRITE “The content of sack is:” , content
WRITE “The weight of sack is:” , weight
ELSE
WRITE “The Content is Invalid”
ENDIF
total_weight ← total_weight + weight
NEXT
WRITE “Total weight of order is:” , total_weight
WRITE “Number of rejected sacks are:” , rej

actual_price ← (c_sack * 3) + (g_sack * 2) + (s_sack * 2)
REPEAT
IF c_sack >= 1 And g_sack >= 2 And s_sack >= 2
THEN
sp ← sp + 1
c_sack ← c_sack - 1
g_sack ← g_sack -2
s_sack ← s_sack - 2
ELSE
WRITE “Order is not a special pack”
ENDIF
UNTIL c_sack = 0  or  g_sack = 0  or  s_sack = 0
```

```
IF sp >= 1
THEN
sp_price ← sp_price * 10
WRITE “Total special packs are” , sp
WRITE “Total price of special packs in dollars”, sp_price
discount_price ← (c_sack * 3) + (g_sack * 2) + (s_sack * 2) + sp_price
WRITE “ The actual price of order is”, actual_price
WRITE “The discounted price of order is” , discount_price
total_discount ← actual_price - discount_price
WRITE “Total discount in dollars for the order is” , total_discount
ELSE
WRITE “Price of regular order is” , actual_price
ENDIF
```