

1 A program is being developed to meet a particular customer requirement.

(a) The program contains these variables:

Variable	Data type
MyChar	CHAR
MyString	STRING
MyInt	INTEGER
MyDOB	DATE

Complete the table by filling in the gaps using functions and/or operators from the **insert**.

Each expression must be valid.

Expression
MyString ← 'X' MyString
MyChar ← ("ABCD", ,)
MyString ← (..... (MyDOB))
MyInt ← (..... (MyString) / 2)

[4]

(b) Different test methods will be used at different stages of the program development.

Complete the table by identifying the test method that matches the test description.

The first row has been completed for you.

Test description	Test method
carried out as soon as a program module has been coded	alpha
carried out as program modules are being combined	
completed by the developers without referring to the code	
completed by the customer	

[3]

(c) During the alpha testing stage, an Integrated Development Environment (IDE) is used to help locate an error that has been identified. The IDE report window feature is used to examine the values assigned to variables.

Explain how **two** other IDE features are used together with the report window feature to help locate the error.

.....

.....

.....

.....

.....

.....

..... [3]

1 (a) The table contains pseudocode examples.

Each example may contain statements that relate to one or more of:

- selection
- iteration (repetition)
- subroutines (procedures or functions).

Complete the table by placing **one or more** ticks (✓) in each row.

Pseudocode example	Selection	Iteration	Subroutine
IF Status = FALSE THEN FOR Count ← 1 TO 20 CALL Reset(Count) NEXT Count ENDIF			
OTHERWISE : Status ← TRUE			
WHILE AllDone() = TRUE			
30 : NextChar ← 'X'			

[4]

(b) Complete the table by giving the appropriate data type.

Variable	Example data value	Data type
Result	5.42	
MonthLetter	"JFMAMJJASOND"	
Birthday	15/11/2009	

[3]

(c) Evaluate each expression in the table by using the data values shown in **(b)**.

Write 'ERROR' if the expression contains an error.

Expression	Evaluates to
INT(Result) + 1 > 6	
NUM_TO_STR(LENGTH(MonthLetter))	
NUM_TO_STR(Result + "3.2")	
MID(MonthLetter, MONTH(Birthday) - 2, 1)	

[4]

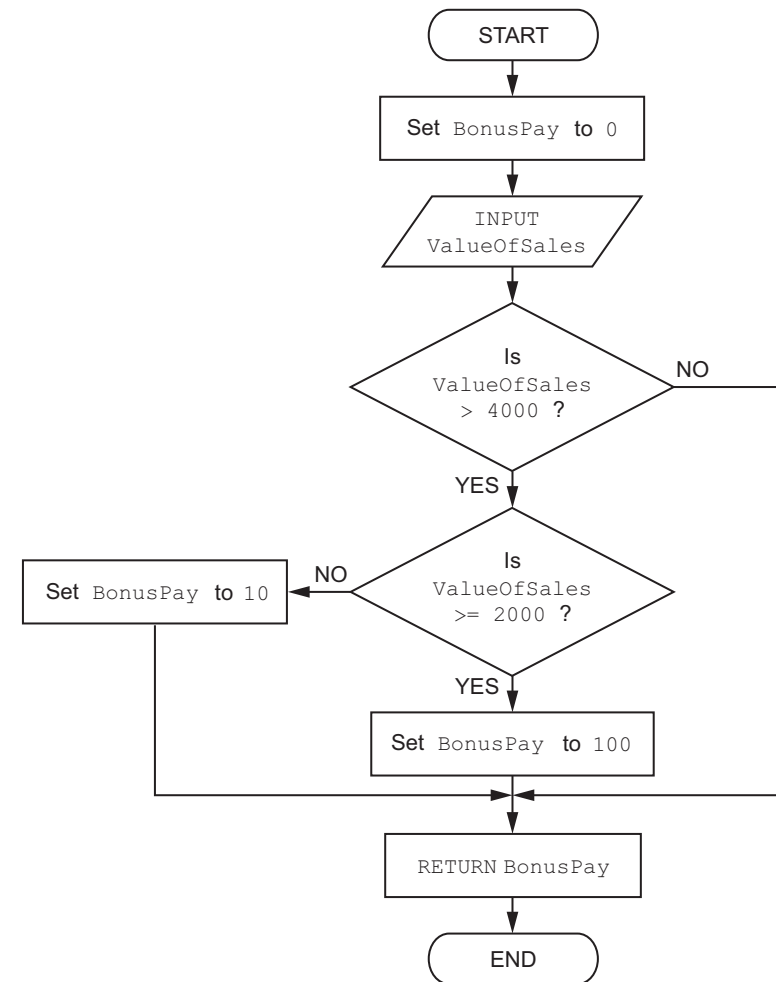
2 A program is being developed to calculate the pay of employees working for a company.

A function `CalculateBonus()` calculates bonus pay based on the value of sales.

Bonus pay is calculated as shown in the table.

Value of sales (in dollars)	Bonus pay (in dollars)
below 2000	0
between 2000 and 4000 inclusive	10
above 4000	100

A flowchart for the function `CalculateBonus()` has been designed.



(a) The flowchart contains logic errors.

One logic error is that `BonusPay` will **not** be set to 10 although the `ValueOfSales` input is between 2000 and 4000 inclusive.

(i) Explain why this error occurs.

.....
 [1]

(ii) Explain how this error could be corrected.

.....
 [1]

(b) There are different ways to reduce the risk of errors when developing the new program, such as the use of constants.

(i) State a value that could be replaced by a constant in the function `CalculateBonus()`

..... [1]

(ii) Explain how the use of constants helps to reduce the risk of programming errors.

.....

 [2]

(iii) One other way that can reduce the risk of errors when writing the program is the use of library routines.

Explain how library routines can reduce the risk of programming errors.

.....
 [1]

(c) All the logic errors have been corrected, and the program has been coded.

Identify and describe **two** other types of error that the program could contain.

Type of error

Description

.....

.....

Type of error

Description

.....

.....

[4]

1 A program will calculate the tax payable based on the cost of an item.

Calculations will occur at many places in the program and these involve the use of one of three tax rates.

Tax rate values represent a percentage. For example, a tax rate value of 5.23 represents 5.23%. In this case, the tax payable on an item costing \$100 would be \$5.23.

Tax rate values are used at several places within the program. One example is given in pseudocode as follows:

```
HighRate ← FALSE
CASE OF ItemCost
    <= 50 : TaxRate ← 3.75      // tax rate of 3.75%
    <= 200 : TaxRate ← 5.23    // tax rate of 5.23%
    > 200 : TaxRate ← 6.25     // tax rate of 6.25%
        HighRate ← TRUE
ENDCASE
TaxPayable ← ItemCost * TaxRate // tax payable
```

(a) The pseudocode contains a logical error.

Identify the error **and** suggest a correction.

Error

.....

Correction

.....

[2]

(b) During the design of the program, tax rate values have been used wherever they are needed as shown in the pseudocode example above. Tax rates do not change while the program runs.

(i) Identify a more appropriate way of representing the tax rate values in the final program.

..... [1]

(ii) Describe the benefits of your answer to part (b)(i) with reference to this program.

.....

.....

.....

.....

.....

..... [3]

(c) Give the **appropriate** data type for the variables in the following table, as used in the pseudocode:

Variable name	Data type
HighRate	
TaxPayable	

[2]

(d) The final CASE condition (> 200) in the pseudocode example could be replaced with a keyword.

Give the keyword.

..... [1]

1 (a) The following table contains pseudocode examples.

Each example may contain statements that relate to one or more of the following:

- selection
- iteration (repetition)
- subroutine (procedure or function).

Complete the table by placing **one or more** ticks (✓) in each row.

Pseudocode example	Selection	Iteration	Subroutine
FOR Index $\leftarrow$ 1 TO 3 IF Safe[Index] = TRUE THEN Flap[Index] $\leftarrow$ 0 ENDIF NEXT Index			
CASE OF Compound(3)			
REPEAT UNTIL AllDone() = TRUE			
WHILE Result[3] <> FALSE			

[4]

(b) Complete the table by giving the appropriate data type in each case.

Variable	Example data value	Data type
Available	TRUE	
Received	"18/04/2021"	
Index	100	

[3]

(c) Evaluate each expression in the table by using the data values shown in part (b).

Write 'ERROR' if the expression contains an error.

Expression	Evaluates to
Available AND NOT(Index > 100)	
Index MOD 30	
NUM_TO_STR(Index + "33")	

[3]

6 In some countries, on the third Sunday in March, daylight saving time begins when clocks move forward by one hour.

A module `AdjustClock()` will take an integer parameter representing a year. The module will return an integer value representing the number of the day in March on which the clocks move forward.

For example, the following line of pseudocode would assign `DayNumber` the value 20:

```
DayNumber ← AdjustClock(2022)
```

Write pseudocode for the function `AdjustClock()`.

Date functions from the **insert** should be used in your solution.

[7]

[7]

1 An algorithm is developed in pseudocode before being coded in a programming language.

(a) The following table shows four valid pseudocode assignment statements.

Complete the table by giving an appropriate data type to declare each of the variables A, B, C and D.

Assignment statement	Data type
A ← LEFT(MyName, 1)	
B ← Total * 2	
C ← INT(ItemCost) / 3	
D ← "Odd OR Even"	

[4]

(b) Other variables in the program have example values as shown:

Variable	Value
Sorted	False
Tries	9
ID	"ZGAC001"

Complete the table by evaluating each expression, using the example values.

Expression	Evaluates to
Tries < 10 AND NOT Sorted	
Tries MOD 4	
TO_LOWER(MID(ID, 3, 1))	
LENGTH(ID & "xx") >= Tries	

[4]

(c) The variable names A, B, C and D in part **(a)** are **not** good programming practice.

(i) State why these variable names are **not** suitable.

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

(ii) Identify **one** problem that these variable names might cause.

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

(iii) The choice of suitable variable names is one example of good programming practice.

Give **one other** example.

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

1 (a) The following table contains pseudocode examples.

Each example may contain statements that relate to one or more of the following:

- selection
- iteration (repetition)
- input/output.

Complete the table by placing **one or more** ticks (✓) in each row.

Pseudocode example	Selection	Iteration	Input/Output
FOR Index ← 1 TO 10 Data[Index] ← 0 NEXT Index			
WRITEFILE ThisFile, "*****"			
UNTIL Level > 25			
IF Mark > 74 THEN READFILE OldFile, Data ENDIF			

[4]

(b) Program variables have data types as follows:

Variable	Data type
MyChar	CHAR
MyString	STRING
MyInt	INTEGER

Complete the table by filling in each gap with a function (from the **insert**) so that each expression is valid.

Expression
MyInt ← (3.1415926)
MyChar ← ("Elwood", 3, 1)
MyString ← (..... (27.509))
MyInt ← (..... ("ABC123", 3))

[4]

(c) The variables given in part **(b)** are chosen during the design stage of the program development life cycle.

The choices are to be documented to simplify program maintenance.

State a suitable way of documenting the variables **and** give **one** piece of information that should be recorded, in addition to the data type.

.....

.....

..... [2]