AQA Computer Science A-Level

4.2.3 Stacks

Past Paper Questions

June 2011 Comp 3

5 (c) The pseudo-code algorithm in **Figure 3** can be used to calculate the result of evaluating a Reverse Polish Notation expression that is stored in a string. The algorithm is designed to work only with the single digit denary numbers 0 to 9. It uses procedures and functions listed in **Table 1**, two of which operate on a stack data structure.

Figure 3

```
StringPos ← 0
Repeat
  StringPos ← StringPos + 1
  Token ← GetCharFromString(InputString, StringPos)
  If Token = '+' Or Token = '-' Or Token = '/' Or Token = '*'
    Then
      Op2 ← Pop()
      Op1 ← Pop()
      Case Token Of
        '+': Result ← Op1 + Op2
        '-': Result ← Op1 - Op2
        '/': Result ← Op1 / Op2
        '*': Result ← Op1 * Op2
      EndCase
      Push(Result)
    Else
      IntegerVal ← ConvertToInteger(Token)
      Push(IntegerVal)
  EndIf
Until StringPos = Length(InputString)
Output Result
```

Table 1

Procedure/Function	Purpose	Example(s)
GetCharFromString (InputString: String, StringPos: Integer): Char	Returns the character at position StringPos within the string InputString. Note that the leftmost letter is position 1, not position 0.	GetCharFromString ("Computing", 1) would return the character 'C'. GetCharFromString ("Computing", 3) would return the character 'm'.
ConvertToInteger (ACharacter: Char): Integer	Returns the integer equivalent of the character in ACharacter.	ConvertToInteger('4') would return the integer value 4.
Length(AString: String): Integer	Returns a count of the number of characters in the string AString.	Length("AQA") would return the integer value 3.
Push(ANumber: Integer)	Puts the number in ANumber onto the stack.	Push(6) would put the number 6 on top of the stack.
Pop(): Integer	Removes the number from the top of the stack and returns it.	X ← Pop() would remove the value from the top of the stack and put it in X.

5 (d) A programmer is going to implement the algorithm from **Figure 3** in a programming language that does not provide built-in support for a stack data structure.

The programmer intends to simulate a stack by using a fixed length array of 20 integers named `StackArray` with indices running from `1` to `20` and an integer variable `TopOfStackPointer` which will be initialised to `0`.

Write a pseudo-code algorithm for the `Push` operation to push a value stored in the variable `ANumber` onto the stack.

Your algorithm should cope appropriately with any potential errors that might occur.

.....

.....

.....

.....

.....

.....

.....

.....

(4 marks)

June 2017 Paper 1

0 6 Two frequently completed actions when using a particular piece of software are **undo** and **repeat**.

The **undo** action results in the state changing from the current state to the state previous to the user's most recent action, eg if the last action the user completed was to change the font of a selected piece of text from `Courier New` to Chiller then if the **undo** action is selected the result will be to change the font of that text back to `Courier New`.

The user is able to keep using the **undo** action to go back through all previous states.

The **repeat** action results in the user's most recent action being applied again, eg if the last action the user completed was to change the font of a piece of text to Chiller then if the **repeat** action is selected the result will be to change the font of the currently selected text to Chiller.

The user is able to keep using the **repeat** action to apply the most recent action multiple times. The **repeat** action will only work when there is a most recent action that can be applied again.

0 6 . 1 Explain how a single stack can be used in the implementation of the repeat action **and** the undo action.

[5 marks]

0 6 . 2 State the type of error that occurs if the user tries to complete an undo action before they have completed any other actions.

[1 mark]