3 A program uses a stack to hold up to 60 numeric values.
The stack is implemented using two integer variables and a 1D array.

The array is declared in pseudocode as shown:

```
DECLARE ThisStack : ARRAY[1:60] OF REAL
```

The stack operates as follows:

- Global variable `SP` acts as a stack pointer that points to the next available stack location. The value of `SP` represents an array index.
- Global variable `OnStack` represents the number of values currently on the stack.
- The stack grows upwards from array element index 1.

(a) (i) Give the initial values that should be assigned to the **two** variables.

`SP` .....

`OnStack` .....
[1]

(ii) Explain why it is **not** necessary to initialise the array elements before the stack is used.

.....

.....

.....

..... [2]

(b) A function to add a value to `ThisStack` is expressed in pseudocode as shown.
The function will return a value to indicate whether the operation was successful or not.

Complete the pseudocode by filling in the gaps.

```
FUNCTION Push(ThisValue : REAL) RETURNS BOOLEAN

   DECLARE ReturnValue : BOOLEAN

   IF ..... THEN

      RETURN ..... // stack is already full

   ENDIF

   ..... ← ThisValue

   SP ← .....

   OnStack ← OnStack + 1

   RETURN TRUE

ENDFUNCTION
```

[4]