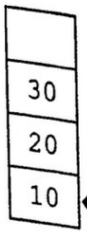
Answer all questions.

1. Peter uses stacks to manage boxes. Each box stores some apples. In the following example, a stack contains 3 boxes with 10, 20 and 30 apples.

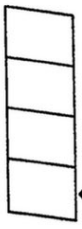


Below are the operations on the stacks:

Operation	Description
`Push(S, k)`	Push a box with `k` apples into stack `S`.
`Pop(S)`	Remove a box from stack `S` and return the number of apples in the box.
`Empty(S)`	Return `TRUE` if stack `S` has no boxes in it; otherwise, return `FALSE`.

(a) (i) Initially there is an empty stack `A`. Write down the final content of `A` after executing the following pseudocode.

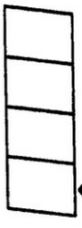
```
Push(A, 10)
Push(A, 20)
TMP ← Pop(A)
if Empty(A) then Push(A, 30)
```



(2 marks)

(ii) Initially there is an empty stack `B`. Write down the final content of `B` after executing the following pseudocode.

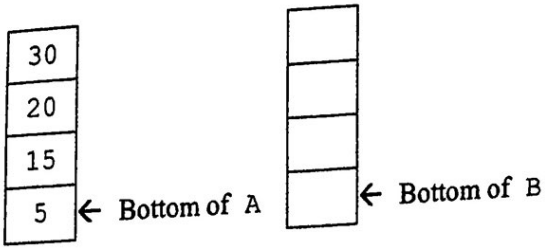
```
Push(B, 10)
Push(B, 20)
Push(B, 30)
Push(B, Pop(B)+Pop(B))
```



(2 marks)

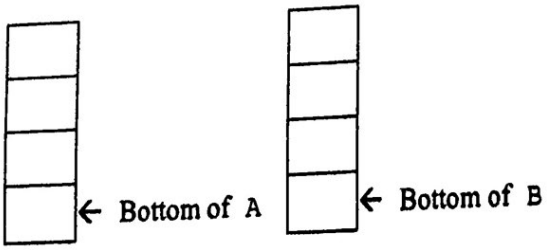
Answers written in the margins will not be marked.

(b) Initially there is a non-empty stack A and an empty stack B, as shown below:



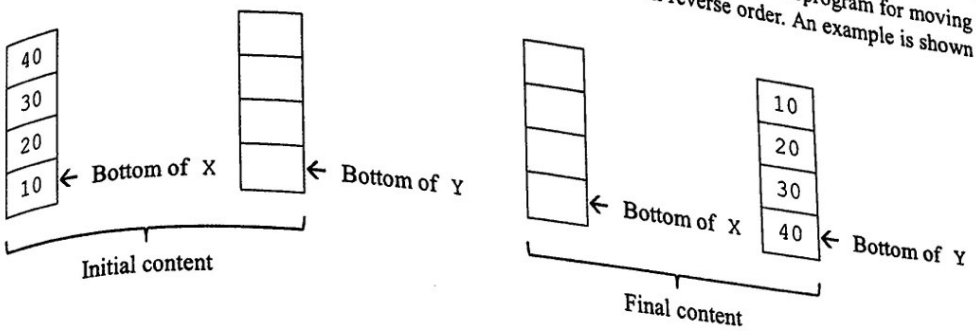
Write down the final content of A and B after executing the following pseudocode.

```
TMP ← 0
while not Empty(A) do
      TMP ← TMP + Pop(A)
      if TMP > 30 then
          Push(B, 30)
          TMP ← TMP - 30
Push(B, TMP)
```



(3 marks)

(c) Initially there is a non-empty stack `X` and an empty stack `Y`. `REV(X,Y)` is a subprogram for moving all the boxes in stack `X` to stack `Y`, where the boxes in `Y` are in reverse order. An example is shown below:



Complete the pseudocode for `REV(X,Y)` below.

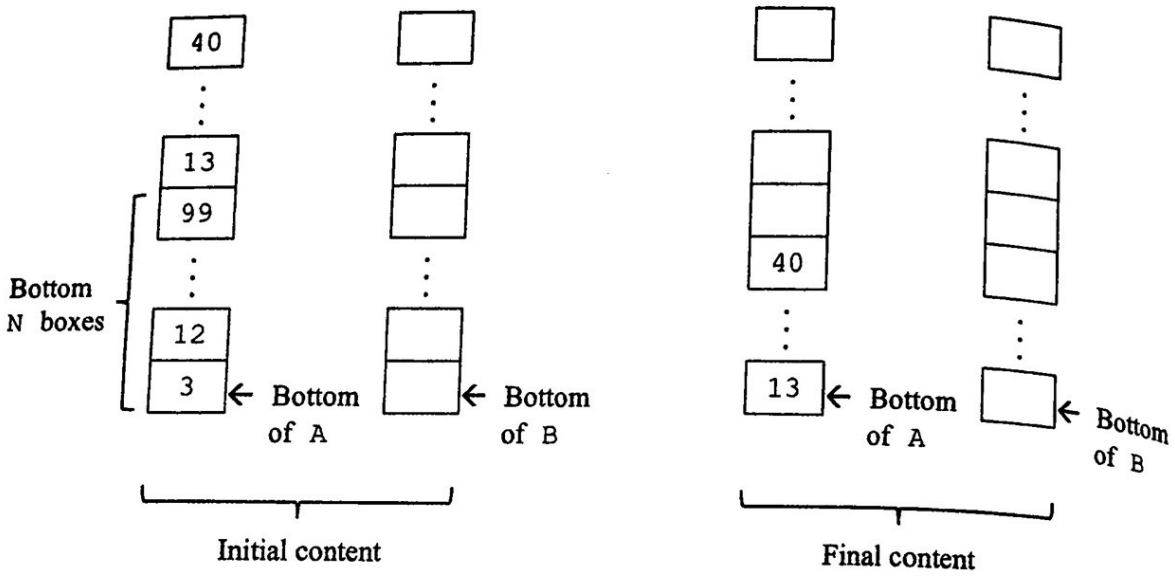
```
REV(X,Y)
```

(3 marks)

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

(d) Initially there is a non-empty stack A and an empty stack B.



It is found that the apples in the bottom N boxes in A are rotten. Write the pseudocode for removing the bottom N boxes and keeping the remaining boxes in the original order in A with the use of REV(X, Y).

(4 marks)

(e) When implementing REV, Peter uses break points for debugging. Describe how break points can help Peter write a program.

(2 marks)

2. Mr Wong plans to write a score processing program. He uses an array `Score` to store `N` students' scores. The scores are sorted in descending order. In the following example, the first seven scores are shown.

Index	1	2	3	4	5	6	7
`Score`	91	83	72	67	67	67	48

A subprogram `QueryByScore(SC)` returns the number of students whose scores are equal to `SC`.

(a) Referring to the example above, what is the return value of `QueryByScore(67)`? _____

(1 mark)

Mr Wong uses the following pseudocode for `QueryByScore(SC)` :

```
QueryByScore(SC)
    i ← BinSearch(SC)
    if i <> -1 then
        a ← goLeft(i)
        b ← goRight(i)
        return b - a + 1
    else return 0
```

where `BinSearch(SC)` returns a value `k` for `Score[k] = SC` using the binary search strategy, or returns `-1` if not found,
`goLeft(i)` returns the smallest value `j` for `Score[j] = Score[i]`, and
`goRight(i)` returns the largest value `j` for `Score[j] = Score[i]`.

(b) (i) Write the pseudocode for `BinSearch(SC)`.

(5 marks)

Answers written in the margins will not be marked.

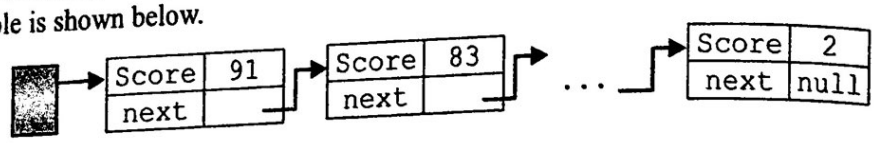
The pseudocode for `goLeft(i)` is

```
goLeft(i)
    j ← i
    while (j > 1) and (Score[j-1] = Score[i]) do
        j ← j-1
    return j
```

(ii) Write the pseudocode for `goRight(i)`.

(3 marks)

(c) Mr Wong considers using a linked list instead of an array to store students' scores in descending orde
An example is shown below.



(i) Mr Wong finds that it is more difficult to implement `goLeft` than `goRight`. Why?

(2 marks)

(ii) Can `BinSearch` be implemented with the linked list efficiently? Explain briefly.

(1 mark)

(iii) Assume that a new top score will be added. Do you agree that it is more efficient to use the linked list instead of the array? Explain briefly.

(2 marks)

3. The system development team of a company redevelops its employee management system. There is some discussion during the system development.

(a) (i) The system development will include five development phases, shown below. In which development phase does each of the following discussions take place?

Phase 1: Systems analysis
Phase 2: Systems design
Phase 3: Systems implementation
Phase 4: Systems conversion and maintenance
Phase 5: Systems documentation

	Development phase (1, 2, 3, 4 or 5)
Discussion 1 Greg: Referring to the Gantt chart, I'll start to write some subprograms next month. Can I have the data flow diagrams? Eva: I'm working on the data flow diagrams based on the collected user requirements. I'll email them to you next week.	
Discussion 2 Eva: The new system has been in use for three months. What have you found? Clara: Some reports generated by the old system and the new system are inconsistent.	
Discussion 3 Tom: I hope that the new system can keep track of the updated employee records. Please design a system with multiple user accounts and implement it after the summer. Eva: Noted. I'll incorporate your requirement into the design.	

(3 marks)

(ii) Who is the system analyst in the system development team? Justify your answer.

(2 marks)

(iii) Give **two** benefits of using a Gantt chart for system development.

(2 marks)

(iv) Referring to Discussion 2, what strategy has been used to change the old system into the new system?

(1 mark

Discussion 4
Greg: What testing will we do?
Eva: We will conduct unit testing, system testing and acceptance testing.

(b) (i) After unit testing is completed, why should system testing be conducted?

(1 mark)

(ii) After system testing is completed, why should acceptance testing be conducted?

(1 mark)

Discussion 5
Greg: In the system, there are a number of subprograms. I suggest using a procedural programming language for the implementation.
Eva: No, we should use an object-oriented programming language to implement the system.

(c) (i) Give an advantage of Greg's suggestion over Eva's.

(1 mark)

(ii) Give an advantage of Eva's suggestion over Greg's.

(1 mark)

(iii) Usually linkers and loaders are involved in compiling an object-oriented program. What are the differences between them?

(2 marks)

Answers written in the margins will not be marked.

4. A grid with 5×6 cells is used to overlay a map showing an island and an ocean, as shown below:

R

i \ j	1	2	3	4	5	6
1	0	1	2	2	2	1
2	0	1	3	7	5	1
3	0	0	2	9	2	0
4	1	2	4	4	3	2
5	1	1	1	0	1	1

In each cell, there is a number representing the number of people (in thousands) living in that area. A two-dimensional array `R` is defined and `R[i,j]` stores the number of people in the corresponding cell.

Peter and Mary plan to build a squared WiFi zone to cover the island. A WiFi zone with `K × K` cells can be represented by `Z(i,j,K)`, where `[i,j]` is the top left corner of the WiFi zone on the map.

(a) Suppose that there is a WiFi zone with `2 × 2` cells.

(i) What is the number of people in `Z(1,2,2)` indicated by the bold square in the grid above?

_____

(1 mark)

(ii) The WiFi zone is relocated to serve the maximum number of people on the map.

(1) The WiFi zone is `Z(`_____`,`_____`, 2)`.

(1 mark)

(2) How many people are living in the WiFi zone? _____

(1 mark)

Answers written in the margins will not be marked.

Answers written in th

Peter develops a subprogram `SumR(i,j,K)` to return the number of people living in the WiFi zone `Z(i,j,K)`.

(b) (i) Complete the pseudocode for `SumR(i,j,K)` below.

```
Line 10:  SumR(i, j, K)
Line 20:      sum ← 0
Line 30:      for a from 1 to [          ] do
Line 40:          for b from 1 to [          ] do
Line 50:              sum ← sum + R[ [          ] , [          ] ]
Line 60:      return [          ]
```

(4 marks)

(ii) Peter finds that `SumR` does not work properly if part of the WiFi zone lies outside the map, for example, `Z(1,5,3)`:

R

i \ j	1	2	3	4	5	6
1	0	1	2	2	2	1
2	0	1	3	7	5	1
3	0	0	2	9	2	0
4	1	2	4	4	3	2
5	1	1	1	0	1	1

Rewrite Line 50 in (b)(i) to solve the problem. Assume that the number of people living outside the grid is zero.

There is another array `S` such that `S[i,j]` stores the number of people living in the rectangular area from `R[1,1]` to `R[i,j]`. For example,

S

i \ j	1	2	3	4	5	6
1	0	1	3	5	7	8
2	0	2	7	16	23	25
3	0	2		27	36	38
4	1	5	16	38	50	54
5	2	7	19	41	54	

R

i \ j	1	2	3	4	5	6
1	0	1	2	2	2	1
2	0	1	3	7	5	1
3	0	0	2	9	2	0
4	1	2	4	4	3	2
5	1	1	1	0	1	1

`S[2,3] = R[1,1] + R[1,2] + R[1,3] + R[2,1] + R[2,2] + R[2,3] = 7`

(c) What is the value in `S[3,3]`? _____ (1 mark)

Instead of summing up all `R` terms, Mary calculates `S[i,j]` by using its adjacent `S` values.

(d) Complete the following formula for `S[5,6]`.

`S[5,6] = R[5,6] + S[5,5] + S[4,6] - S[` ☐ `,` ☐ `]` (2 marks)

Mary then develops a subprogram `SumS(i,j,K)` to return the number of people living in the WiFi zone `Z(i,j,K)` by using `S`.

(e) Complete the following formula for calculating `Z(3,4,2)` in `SumS`.

```
Z(3,4,2) = S[4,5] - S[4,3] - S[2,5] + S[ ,  ]
```

(2 marks)

(f) For a grid with a very large number of cells, why is Mary's method (`SumS`) better than Peter's method (`SumR`)?

(2 marks)

END OF PAPER