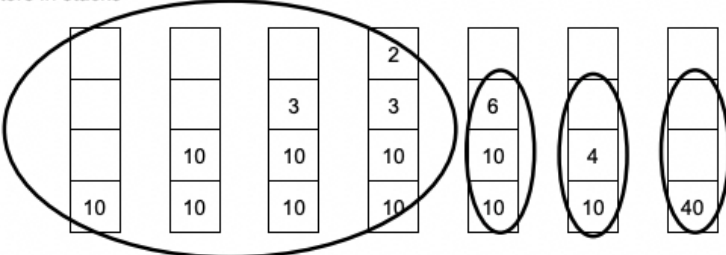


Question	Answer	Marks														
3(a)	<p>One mark for each correct line from Operating System Term to Description</p> <table><thead><tr><th>OS term</th><th>Description</th></tr></thead><tbody><tr><td>Multi-tasking</td><td>Using secondary storage to simulate additional main memory</td></tr><tr><td>Paging</td><td>Managing the processes running on the CPU</td></tr><tr><td>Interrupt handling</td><td>Managing the execution of many programs that appear to run at the same time</td></tr><tr><td>Scheduling</td><td>Locating non-contiguous blocks of data and relocating them</td></tr><tr><td>Virtual memory</td><td>Transferring control to another routine when a service is required</td></tr><tr><td></td><td>Reading/writing same-size blocks of data from/to secondary storage when required</td></tr></tbody></table>	OS term	Description	Multi-tasking	Using secondary storage to simulate additional main memory	Paging	Managing the processes running on the CPU	Interrupt handling	Managing the execution of many programs that appear to run at the same time	Scheduling	Locating non-contiguous blocks of data and relocating them	Virtual memory	Transferring control to another routine when a service is required		Reading/writing same-size blocks of data from/to secondary storage when required	5
OS term	Description															
Multi-tasking	Using secondary storage to simulate additional main memory															
Paging	Managing the processes running on the CPU															
Interrupt handling	Managing the execution of many programs that appear to run at the same time															
Scheduling	Locating non-contiguous blocks of data and relocating them															
Virtual memory	Transferring control to another routine when a service is required															
	Reading/writing same-size blocks of data from/to secondary storage when required															
3(b)	<p>One mark for each correct statement (Max 4)</p> <ul style="list-style-type: none">• An interpreter examines source code one statement at a time• Check each statement for errors• ...If no error is found the statement is executed• ...If an error is found this is reported and the interpreter halts• Interpretation is repeated for every iteration in repeated sections of code/in loops• Interpretation has to be repeated every time the program is run	4														

Question	Answer	Marks
4(a)(i)	<p>One mark for each correct marking point (Max 2)</p> <ul style="list-style-type: none"> • Reverse Polish Notation provides an unambiguous method of representing an expression • ... reading from left to right • ...without the need to use brackets • ...with no need for rules of precedence / BODMAS 	2

Question	Answer	Marks
4(a)(ii)	<p>One mark for identification of the data structure, One mark for a sensible reason</p> <p>Either: Structure: stack The operands are popped from the stack in the reverse order to how they were pushed</p> <p>Or: Structure: Binary tree A (binary) tree allows both infix and postfix to be evaluated (tree traversal)</p>	2
4(b)	$a \ b - \ a \ c + \ * \ 7 \ /$	1
4(c)	$a \ / \ b \ * \ 4 - (a + b)$	1
4(d)	<p>1 mark for correct structure 1 mark for correct substitution</p> <p>$(a + b) \ / \ (c \ / \ d)$ $(17 + 3) \ / \ (48 \ / \ 12)$</p>	2

Question	Answer	Marks
4(a)	<p>One mark for each marking point (Max 2)</p> <ul style="list-style-type: none"> • <code><character> ::=</code> • <code>\$ % & * #</code> <p>Complete answer <code><character> ::= \$ % & * #</code></p>	2
4(b)(i)	For example: \$A9E3	1
4(b)(ii)	<p>One mark for each marking point (Max 4)</p> <ul style="list-style-type: none"> • <code><password> ::= <character> ...</code> • <code>... <code></code> • <code><code> ::= ...</code> • <code>... <digit> <capital_letter></code> • <code>... <digit><code> <capital_letter><code></code> <p>Complete answer <code><password> ::= <character><code></code></p> <p><code><code> ::= <digit> <capital_letter> <digit><code> <capital_letter><code></code></p>	4

Question	Answer	Marks
5(a)	<p>One mark for each in order $j\ k+j\ k- /$ $j\ k+$ $j\ k- /$</p>	2
5(b)(i)	<p>1 mark per ring Do not allow operators in stacks</p> 	4
5(b)(ii)	<p>Any four from Max 4 Max 3 generic answer only</p> <ul style="list-style-type: none"> Working from left to right in the expression PUSH 10/m onto the stack PUSH the following numbers (10/m, 3/j, 2/k) onto the stack When the first operator, *, is reached ... POP the top two numbers, 2/k and 3/j ... apply the operation PUSH result back onto stack Continue to the end of the expression 	4
5(c)	<p>Any two from</p> <ul style="list-style-type: none"> recursion implementation of ADTs e.g. linked lists procedure calls interrupt handling (storing contents of registers etc) 	2

Question	Answer	Marks
4	<p>One mark for each correct line connecting one stage of compilation to a description</p> <p>Stage of compilation</p> <p>Description</p>	4

Question	Answer	Marks
5(a)	$a \cdot b + d - 15 +$	1
5(b)(i)	$(a - b) \cdot (c + d) / a$	1
5(b)(ii)	-39	1