Skema Jawapan

<table>
<tr><th>No Soalan</th><th>Jawapan</th><th>Markah</th></tr>
<tr><td>1 (a)(i)</td><td>1. Testing
2. Testing ensure the program works correctly and fixes errors (bugs)</td><td>1m
1m</td></tr>
<tr><td>1 (a)(ii)</td><td>1. Problem Analysis
2. Problem analysis is the act of identifying input, process and output (IPO).</td><td>1m
1m</td></tr>
<tr><td>1 (b)</td><td><table><tr><td>Design a Solution</td><td>Design a solution involves creating an algorithm derived from the input, process and output of the problem analysis step.</td></tr><tr><td>Documentation</td><td>A detailed write-up explaining how the program works</td></tr></table></td><td>0.5 m for each box.

* The explanation can only be given marks if the steps is correct</td></tr>
<tr><td>1 (c)</td><td>The Input-Process-Output (IPO) Model is an approach to describe and visualize the input, process and output to solve the given problem</td><td>1m</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>1 (d)</td><td>1. Process
2. Input
3. Output</td><td>1m
1m
1m</td></tr>
</table>

1 (e)	Input : S1, S2,S3 Process : Calculate the total surface area and total cost based on S1,S2,S3 and cost per m2 Output : total surface area and total cost	1m for input 2m for process 1m for output
1 (f)	Input :position, monthly salary Process : Determine “Staff” or “Manager” or “Error” message based on position Calculate and determine the tax rate, tax amount and net salary after tax deduction based on position and monthly salary Output : “Error” Or “Staff” or “Manager” and monthly salary, tax rate, tax amount, and net salary after tax deduction	1m 3m 2m
1 (g)	Input : amount paid for 5 times Process : repeat calculates the amount of petrol purchased based on amount paid and price per litre for 5 times Output : Amount of petrol purchased for 5 times	1m 2m 1m

<table>
<tr>
<td>2
(a)</td>
<td>START

if mark < 40
Display “Fail”
else
Display “Pass”
end if

END</td>
<td>0.5m for if

0.5m for else

1m for both print

* marks for print only given if the print message is in the correct if/else</td>
</tr>
<tr>
<td>2
(b)</td>
<td></td>
<td>1m for while n < 30 or n < 31
1m for if temp > max temp
1m for correct average calculation
1m for input inside loop
3m for print max temp, date of max temp and average</td>
</tr>
<tr>
<td>2
(c)</td>
<td>
<table>
<tr><th>Pseudocode</th><th>Flowchart</th></tr>
<tr><td>Informal language using English like language to design algorithms</td><td>A graphical representation of a algorithm in relation to its sequence of functions</td></tr>
</table>
</td>
<td>1m for each box

* Marks for differentiate is only given if the reasoning match the technique</td>
</tr>
</table>

3 (a) (i)	Procedural : Logic :	
3 (a) (ii)	Assembler : A program that translates assembly language into machine language. Interpreter: A program that translates translate the program's high level instructions line by line into machine language instructions as the program is running	1m 1m
3 (a) (iii)	1. Invalid 2. valid	1m 1m
3 (a) (iv)	Runtime error	1m
3 (a) (v)	They are reserved keywords	1m
3 (a) (vi)	1. Float/ Double 2. Boolean	1m 1m
3 (a) (vii)	1. Selection 2. Repetition	1m 1m
3 (a) (viii)	1. sphere_volume = (4/3) * 3.142 * r* r 2. status = 'Y'	1m 1m
3 (a) (ix)	1. Relational 2. Arithmetic 3. Boolean	1m 1m 1m

<table>
<tr><td>4
(a)
(i)</td><td>Circumference: 188.52
Around the circle</td><td>1m</td></tr>
<tr><td>4
(a)
(ii)</td><td>Circumference: 0
Around the circle</td><td>1m</td></tr>
<tr><td>4
(b)</td><td>1
4
4</td><td>1m
1m
1m</td></tr>
<tr><td>4
(c)</td><td><table>
<tr><td>loop that executes a specific number of times.</td><td>for i in range (number):
statement</td></tr>
<tr><td>A sentinel-controlled loop uses a sentinel value to stop a loop</td><td>Prime input
while condition
statement
Update input</td></tr>
</table></td><td>1m for each box</td></tr>
<tr><td>4
(d)</td><td>total = 0
for i in range (2,11,2):
print(i)
total += i
print(total)</td><td>0.5 m input

2m for loop

0.5 m print

0.5 add total

0.5m print</td></tr>
</table>

4
(e)
	total_even = 0
	count_even = 0
	for i in range(250):
	  number = float(input(“Enter number”))
	  if number % 2 == 0:
	  total_even += number
	  count_even += 1
	
	if count_even > 0:
	  average = total_even/count_even
	  print (average)
	else:
	  print (0)
		0.5m each var declare
		
		1m for with correct range and input
		
		1m for correct if
		
		1m for correct total_even addition
		1m for correct count_even addition
		
		1m for correct if , average calculator and print
		
		0.5 for correct else and print

5	for i in range (51):
	  name = input("Enter name")
	  marks = float(input("Enter marks"))
	
	  if marks >= 80 and marks <= 100:
	  print("A")
	  print(name)
	  elif marks >= 70 and marks < 80:
	  print("B")
	  elif marks >=60 and marks < 70:
	  print("C")
	  elif marks >= 40 and marks < 60:
	  print("D")
	  elif marks > 0 and marks < 40:
	  print("F")
	  print("Name")
	  else:
	  print(“Error”)
		1m for loop
		
		0.5m input name
		
		0.5m input marks
		
		1m for each correct if condition and print
		
		0.5m for print name in grade A
		
		0.5m for print name in grade F

6	divisible_by_five_only = 0
	divisible_by_seven_only = 0
	divisible_by_both = 0
	total = 0
	count = 0
	
	number = float(input("Enter a positive number (negative to quit): "))
	
	while number >= 0:
	  if number % 7 == 0 and number % 5 == 0:
	  divisible_by_both += 1
	  elif number % 7 == 0:
	  divisible_by_seven_only += 1
	  elif number % 5 == 0:
	  divisible_by_five_only += 1
	
	  total += number
	  count += 1
	
	  number = float(input("Enter a positive number (negative to quit): "))
	
	print("Numbers divisible by 5 only:", divisible_by_five_only)
	print("Numbers divisible by 7 only:", divisible_by_seven_only)
	print("Numbers divisible by both 5 and 7:", divisible_by_both)
	
	if count > 0:
	  average = total / count
	  print("Average of all numbers:", average)
	else:
	  print("Average of all numbers: 0")
		0.5m - Counters initialized (divisible_by_five_only, divisible_by_seven_only, divisible_by_both)
		0.5m - variables initialized (total, count)
		1m - Initial input before loop
		1m - While loop with correct condition >= 0
		0.5m - Input inside loop
		
		1m - Check both 5 AND 7 first
		1m - Check 7 only with elif
		1m - Check 5 only with elif
		
		* Order of if/elif is very important, see me if different
		
		0.5m - Accumulate total
		0.5m - Count numbers
		0.5m - Print divisible by 5 only
		0.5m - Print divisible by 7 only
		0.5m - Print

		divisible by both **0.5m** - Calculate average **0.5m** - Print average Deduct 0.5 m for each message that is not exactly like in the question 0.5m for print with the wrong order