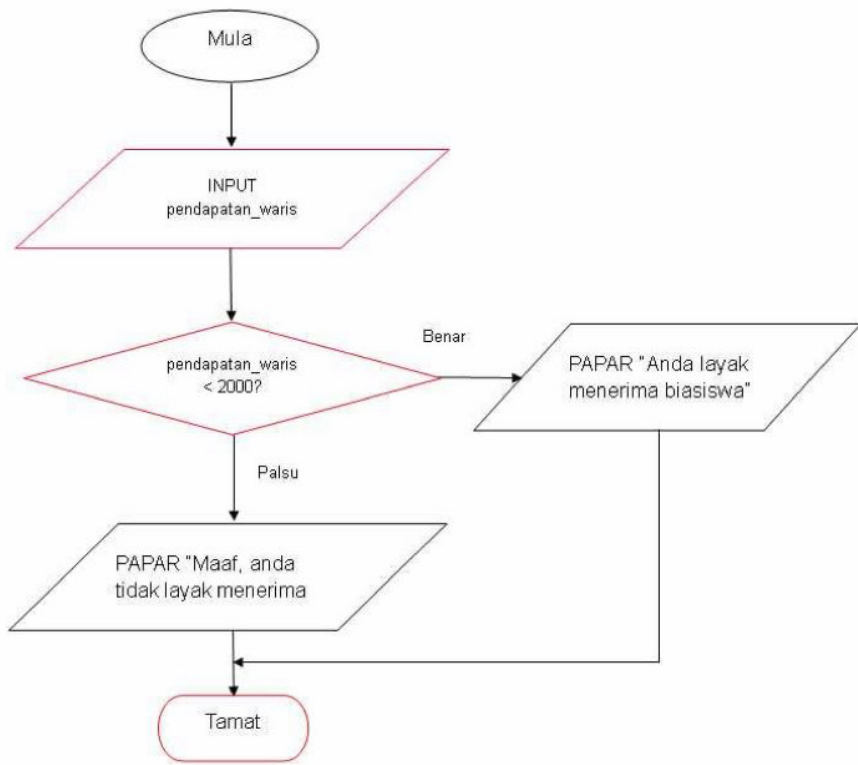
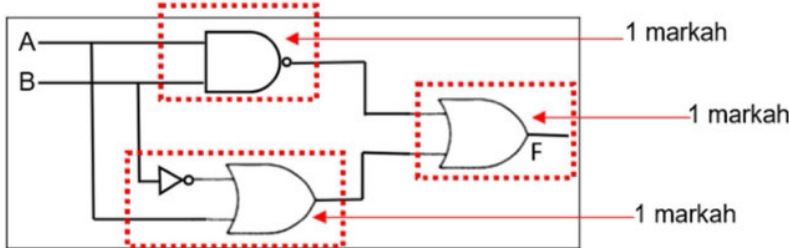


PERATURAN PEMARKAHAN

NO	JAWAPAN	MARKAH
1.	A : Kos B: Sumber	1m 1m
2.	(a) $i \leq 100$ (b) $i = i + 1 / i += 1 / i++$ (c) papar/cetak jumlah	1m 1m 1m
3.	a) i) 15 ii) 22 b) Perimeter: 22	1m 1m 1m
4.	 <pre> graph TD Mula([Mula]) --> Input[/INPUT
pendapatan_waris/] Input --> Decision{pendapatan_waris
< 2000?} Decision -- Benar --> Papar1[/PAPAR "Anda layak
menerima biasiswa"/] Decision -- Palsu --> Papar2[/PAPAR "Maaf, anda
tidak layak menerima"/] Papar1 --> Tamat([Tamat]) Papar2 --> Tamat </pre>	1m 1m 1m
5.	i) Ralat logik ii) Ralat masa larian iii) Ralat sintaks	1m 1m 1m

6.	i) Fungsi ▪ Mengembalikan data / memulangkan data ▪ Badan diakhiri dengan pernyataan return diikuti data yang dipulangkan ▪ Jenis data pulangan dalam sintaks seperti int, double, char, string dan objek java.	1m
	ii) Prosedur ▪ Tidak mengembalikan data / memulangkan data ▪ Badan tidak diakhiri dengan perkataan return ▪ Jenis data pulangan dalam sintaks adalah void ** mana – mana jawapan boleh diterima / pilih salah satu	1m
7.	X : Fasa Analisis Masalah	1m
	Y : Fasa Laksana Penyelesaian	1m
8.	X ▪ menyokong hubungan banyak entiti induk kepada banyak entiti anak ▪ terdiri daripada beberapa jenis rekod dan dihubungkan melalui penunjuk	1m
	Y ▪ data disusun dalam jadual yang terdiri daripada lajur dan baris ▪ struktur pangkalan data tidak perlu dinyatakan terlebih dahulu ▪ menggunakan kunci primer untuk mengenal pasti setiap rekod secara unik dalam sesuatu jadual ▪ kunci asing digunakan untuk menghubungkan antara jadual	1m
	Z ▪ data disusun dalam struktur pokok ▪ suatu entiti mempunyai hubungan satu induk (parent) dengan satu atau lebih entiti anak (child) ▪ model ini tidak menyokong hubungan banyak entiti induk kepada banyak entiti anak ** pilih satu ciri sahaja bagi setiap model	1m
9.	P: Kebergantungan fungsi sepenuhnya	1m
	Q : Kebergantungan fungsi transitif	1m



13.	Dua Pengesahan / Double verifcation	1m				
14.	a) Caesar Cipher ** ejaan perlu betul b) SEJAHTERA ** perlu huruf besar	1m 1m				
15.	P : Capai Q: Laksana R: Nyahkod	3m				
16.	Label Input A dan B dan output F = 1 markah 	4m				
17.	(a) $(1800 + 2500 + 3500 + 4500) / 4$ (1m) = 3075 @ 3075.00 (1m) (b) <table border="1" data-bbox="414 1198 675 1294"><tr><td>Jumlah</td><td>(1m)</td></tr><tr><td>2600</td><td>(1m)</td></tr></table>	Jumlah	(1m)	2600	(1m)	2m 2m
Jumlah	(1m)					
2600	(1m)					
18.	Isihan Pilih	1m				
19.	Harga komputer RM3500.00 	2m				
20.	X: Breadcrumbs Y: Menu Drop-down	1m 1m				



BAHAGIAN B

NO	JAWAPAN	MARKAH
1.	<pre>import java.util.Scanner; public class PurataBayaranElektrik { public static void main(String[] args) { Scanner input = new Scanner(System.in); double bayaran, jumlah = 0, purata; // Gelung for untuk 12 bulan for (int bulan = 1; bulan <= 12; bulan++) { System.out.print("Masukkan bayaran elektrik bulan ke-" + bulan + ": RM"); bayaran = input.nextDouble(); jumlah += bayaran; } // Kira purata purata = jumlah / 12; // Paparan output System.out.println("Jumlah keseluruhan bayaran: RM" + jumlah); System.out.println("Purata bayaran bulanan: RM" + purata); input.close(); } }</pre> Mana-mana amalan terbaik pengaturcaraan yang digunakan @ <pre>import java.util.Scanner; public class PurataBayaranElektrik { public static void main(String[] args) { Scanner input = new Scanner(System.in); double bayaran, jumlah = 0, purata;</pre>	1m 3m 1m 1m 1m 1m 1m 1m 1m

int bulan = 1;	1m
<pre> // Gelung while untuk 12 bulan while (bulan <= 12) { System.out.print("Masukkan bayaran elektrik bulan ke-" + bulan + ": RM"); bayaran = input.nextDouble(); jumlah += bayaran; bulan++; // tambah nilai bulan } </pre>	1m 1m 1m 1m
<pre> // Kira purata purata = jumlah / 12; </pre>	1m
<pre> // Paparan output System.out.println("Jumlah keseluruhan bayaran: RM" + jumlah); System.out.println("Purata bayaran bulanan: RM"+ purata); </pre>	1m 1m
<pre> input.close(); } } </pre> Mana-mana amalan terbaik pengaturcaraan yang digunakan @	1m
<pre> import java.util.Scanner; public class PurataBayaranElektrik { public static void main(String[] args) { Scanner input = new Scanner(System.in); double bayaran, jumlah = 0, purata; int bulan = 1; // Gelung do-while untuk 12 bulan do { System.out.print("Masukkan bayaran elektrik bulan ke-" + bulan + ": RM"); bayaran = input.nextDouble(); jumlah += bayaran; </pre>	1m 1m 1m

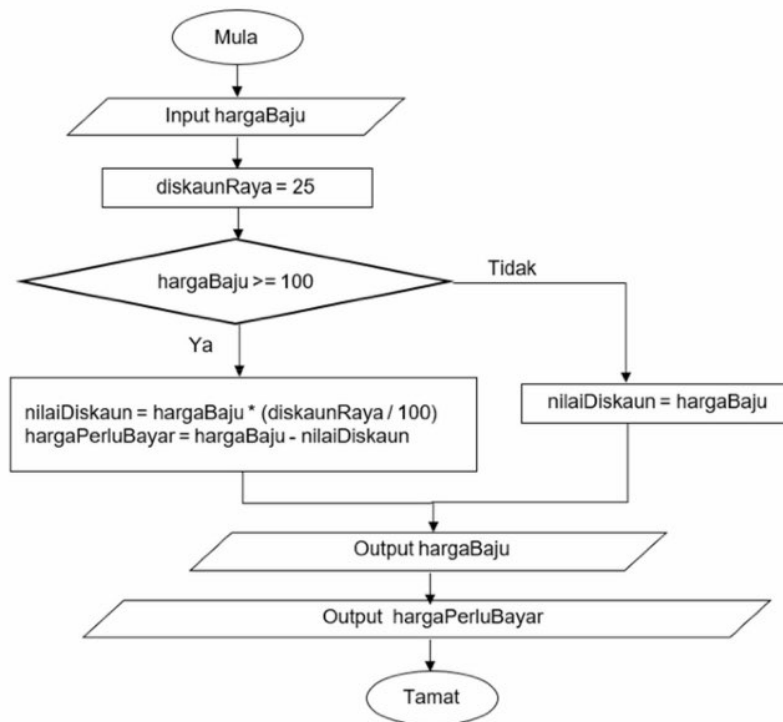
	<pre> bulan++; // tambah nilai bulan } while (bulan <= 12); // Kira purata purata = jumlah / 12; // Paparan output System.out.println("Jumlah keseluruhan bayaran: RM" + jumlah); System.out.println("Purata bayaran bulanan: RM"+ purata); input.close(); } } Mana-mana amalan terbaik pengaturcaraan yang digunakan </pre>	1m 1m 1m 1m 1m 1m 1m
2.	(a) Atur cara B merupakan atur cara yang paling tepat. - Atur cara B menggunakan fungsi prompt() untuk menerima input daripada pengguna, manakala Atur cara A menggunakan fungsi alert() untuk mendapatkan input pengguna. - prompt() digunakan untuk menerima input daripada pengguna / alert() hanya memaparkan mesej. - Atur cara B membahagikan peratus diskaun dengan 100, manakala Atur cara A tidak membahagikan peratus diskaun dengan 100. - Atur cara B lebih tepat kerana peratus diskaun perlu dibahagi dengan 100 sebelum didarab dengan harga asal. - Atur cara B menggunakan syarat pilihan hargaBaju >= 100, manakala Atur cara A menggunakan syarat pilihan hargaBaju > 100. - Atur cara B lebih tepat kerana baju yang berharga RM 100 dan ke atas boleh mendapatkan diskaun. @ Atur cara B menggunakan operator hubungan yang sesuai mengikut syarat bagi mendapatkan diskaun.	1m 1m 1m 1m 1m 1m 1m

- Atur cara B memaparkan nilai pemboleh ubah hargaPerluBayar, manakala Atur cara A memaparkan nilai pemboleh ubah diskaunRaya.
- Atur cara B lebih tepat kerana pemboleh ubah hargaPerluBayar mewakili jumlah akhir yang perlu dibayar selepas diskaun ditolak. / Atur cara B memaparkan nilai yang betul / Atur cara A memaparkan nilai yang salah dan menyebabkan ralat logik.

1m

1m

b)



6m

Input – 1m

3 proses – 2m / kurang 3 proses - 1m

Syarat & Ya / tidak – 1m

2 Output – 2m



SERTAI TELEGRAM @exercise_students

3.

a)

Baris	Sintaks SQL Betul
4	NoTelefon varchar (12),
5	PRIMARY KEY (IDAhli)
8	CREATE TABLE CALON (
12	PRIMARY KEY (IDCalon)
21	FOREIGN KEY (IDAhli) REFERENCES AHLI (IDAhli),
22	FOREIGN KEY (IDCalon) REFERENCES CALON (IDCalon)

**** abaikan nombor baris**

b)

BORANG PENDAFTARAN CALON BAHARU

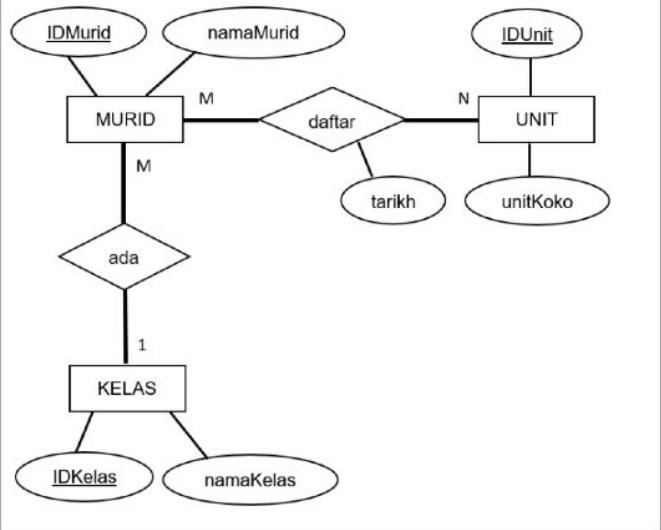
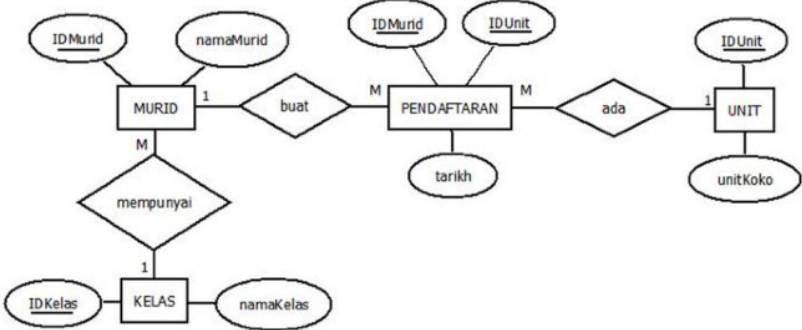
ID CALON*

NAMA CALON

JAWATAN

- Tajuk yang relevan
DAFTAR CALON / PENDAFTARAN CALON BAHARU - **1m**
- Nama medan lengkap - **1m**
- Kotak input lengkap - **1m**
- Butang DAFTAR / SIMPAN - **1m**

1m
1m
1m
1m
1m
1m

4.		8m
	Entiti & Atribut – 1m x 3 entiti Kekardinalan – 2m Hubungan – 1m Kunci Primer – 1m Atribut tarikh pada daftar -1m  Entiti & Atribut – 1m x 4 entiti Kekardinalan – 2m Hubungan – 1m Kunci Primer – 1m	7m



	b) PENDAFTARAN(IDMurid<KP><KA>, IDUnit<KP><KA>, tarikh) MURID (IDMurid<KP>, namaMurid, IDKelas<KA>) UNIT(IDUnit<KP>, unitKoko, lokasi) KELAS(IDKelas<KP>, namaKelas) Kunci Primer bagi entiti MURID, UNIT dan KELAS – 1m Kunci Asing pada entiti MURID – 1m Setiap entiti dan atribut lengkap – 1m x 4 <KP> <KA> pada entiti PENDAFTARAN -1m	
--	--	--