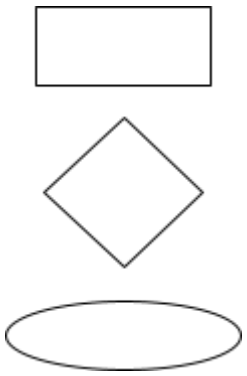
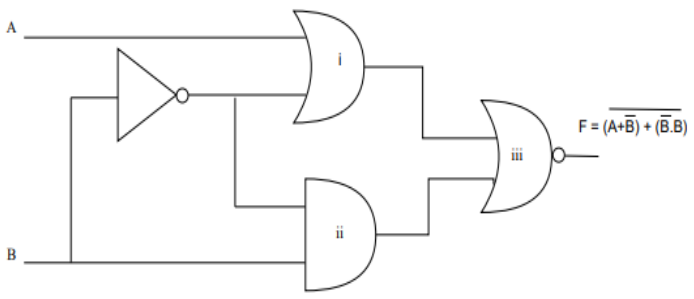
**PANDUAN PEMARKAHAN PERCUBAAN SPM 2025 NEGERI PERAK**

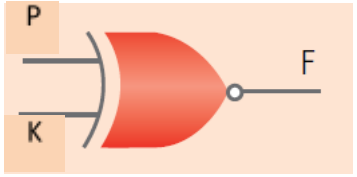
**BAHAGIAN A**

| No Soalan | Cadangan Jawapan | Pemarkahan |
|---|---|---|
| 1 | P : Kos<br>Q : Sumber<br>R : Masa | 1m<br>1m<br>1m |
| 2 | ● **Input**: Algoritma ini menerima input daripada pengguna, iaitu "Masukkan nombor".<br>● **Proses**: Algoritma ini melaksanakan proses pengiraan baki hasil pembahagian nombor dengan 2.<br>● **Output**: Algoritma ini menghasilkan output iaitu memaparkan sama ada nombor tersebut "Nombor Genap" atau "Nombor Ganjil".<br>● Algoritma melibatkan struktur kawalan pilihan. | 1m<br>1m<br>1m<br>1m<br>(Pilih mana-mana tiga jawapan) |
| 3 | Float<br>Boolean | 1m<br>1m |
| 4 | A: do-while<br>B: for<br>C: while | 1m<br>1m<br>1m |
| 5 | Jumlah markah: 210<br>Purata markah: 70.0 | 1m<br>1m |
| 6 | **Kelebihan:**<br>1. **Kod aturcara menggunakan tatasusunan** menggunakan 1 pemboleh ubah dan aturcara menjadi lebih pendek.<br>2. **Kod lebih tersusun dan mudah dibaca** – Setiap modul hanya melaksanakan satu tugasan tertentu.<br>3. **Penggunaan semula kod (reusability)** – Subatur cara boleh dipanggil berkali-kali tanpa menulis semula kod.<br>*1m untuk kelebihan tatasusunan<br>*1m untuk kelebihan submodular | 1m<br>1m |
| 7 | A = Antara muka pengguna bergrafik / Graphical User Interface (GUI) | 1m |

| | | |
|---|---|---|
| 8 |  | 1m<br><br>1m<br><br>1m |
| 9 | i) Tidak boleh mempunyai nilai yang sama secara berulang<br>ii) Mempunyai nilai yang unik<br>iii) Tidak boleh dibiarkan kosong tanpa mengisi data<br>iv) Nilai yang tetap dan tidak berubah | 1m<br>1m<br>1m<br>(pilih 3 sahaja) |
| 10 | i) <br>ii) "Seroja" | 1m<br>1m |
| 11 | i) X: Boleh dipelajari<br>ii) Y: Maklum balas | 1m<br>1m |
| 12 | B<br>D<br>A<br>C | 1m<br>1m<br>1m<br>1m |
| 13 | 2<br>1<br>3 | 1m<br>1m<br>1m |
| 14 | Dua pengesahan (Double verification)<br><br>Lapisan pengesahan tambahan yang perlu ditentukan oleh pengguna apabila ingin memasuki akaun atau mengakses data dalam talian | 1m<br><br>1m |
| 15 | i) Pemproses A<br>ii) 3.2 GHz = 3,200,000,000 Hertz (Hz) | 1m<br>1m |
| 16 |  | i = 1 markah<br><br>ii = 1 markah<br><br>iii = 1 markah |

| | | |
|---|---|---|
| 17 | <table><tr><td>Harga Seunit</td></tr><tr><td>10.30</td></tr></table> | 1m<br>1m |
| 18 | i. Prosedur<br>ii. Fungsi | 1m<br>1m |
| 19 | X = Borang Log Masuk / Login form / Sign-in form<br>Y = Borang Daftar Pengguna /Register form/ Sign-up form | 1m<br>1m |
| 20 | A: Kotak *prompt*<br>B: Kotak *alert*<br>C: Kotak *confirm*<br>D: Butang *radio* | 1m<br>1m<br>1m<br>1m |

**BAHAGIAN B**

| No Soalan | Cadangan Jawapan | Pemarkahan |
|---|---|---|
| 1 | **(a)** Kod atur cara ini membuat **sambungan ke pangkalan data** untuk memperoleh data yang dikehendaki dan **disimpan hasilnya ke dalam pembolehubah laksana**<br><br>Penyataan ini adalah untuk **memaparkan id_pekerja dan nama_pekerja** dalam **jadual pekerja**<br>Yang **berjawatan pengurus**<br>Dan **disusun menurun** mengikut **nama_pekerja**<br><br>(b) (i) COUNT<br><br>(ii) **Select COUNT(*)**<br>AS **Bilangan**<br>**From Pekerja** | 1m<br><br>1m<br><br>1m<br>1m<br>1m<br>1m<br><br>1m<br><br>1m<br>1m<br>1m |
| 2 | a) GET XTAKATAU @GET TAK & GET ATAU & GET DAN<br><br>b) F = $\overline{P \oplus K}$<br>c)<br><br><br>d) | 1m<br><br>1m+1m<br><br>1m<br>1m<br>1m<br><br>1m<br>1m<br>1m |

| | | |
|---|---|---|
| | <table><tr><th>P</th><th>K</th><th>P⊕K</th><th>F = XNOR</th></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>1</td></tr></table><br><table><tr><th>P</th><th>K</th><th>P'</th><th>K'</th><th>P.K</th><th>P'.K'</th><th>(P.K) + (P'.K')</th></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr></table> | 1m |
| 3(a) | Pilihan Terbaik: Atur Cara A<br><br>**Struktur Modular**<br>Atur Cara A menggunakan struktur modular dengan memecahkan kod kepada fungsi kecil seperti paparLuas() dan paparPerimeter(), berbanding Atur Cara B yang menulis semua kod dalam fungsi main() System.<br><br>**Implikasinya:** Kod dalam Atur Cara A lebih mudah diselenggara dan diuji secara berasingan, manakala Atur Cara B lebih sukar untuk diubah suai kerana semua yste bercampur dalam satu tempat.<br><br>**Kebolehbacaan Kod**<br>Atur Cara A lebih kemas dan mudah difahami kerana setiap fungsi mempunyai nama yang jelas menunjukkan tujuannya, berbanding Atur Cara B yang perlu dibaca dari atas ke bawah untuk memahami keseluruhan sistem<br><br>**Implikasinya:** Atur Cara A lebih mesra pembangun (developer-friendly), terutamanya dalam projek besar atau sistem bekerja secara berkumpulan.<br><br>**Kebolehlanjutan (Scalability)**<br>Atur Cara A lebih mudah ditambah baik, contohnya jika ingin tambah fungsi kiraan diameter atau isipadu sfera, fungsi baru boleh dicipta dengan mudah berbanding Atur Cara B yang perlu ditulis semula dalam struktur ulangan yang sama.<br><br>**Implikasinya:** Atur Cara A lebih sesuai digunakan dalam sistem yang memerlukan ciri tambahan atau pengemaskinian dari semasa ke semasa.<br><br>***<br>Mana-mana jawapan yang sesuai dan relevan diterima. Jawapan dalam jadual dan jelas juga diterima. | 1m<br><br>1m<br><br>1m<br><br>1m<br><br>1m<br><br>1m<br><br>1m |

| Aspek | Perbandingan / Huraian | Implikasi |
|---|---|---|
| **Struktur Modular** | Atur Cara A menggunakan struktur modular dengan memecahkan kod kepada fungsi kecil seperti paparLuas() dan paparPerimeter(). Atur Cara B pula menulis semua kod dalam fungsi main(). | Kod dalam Atur Cara A lebih mudah diselenggara dan diuji secara berasingan. Kod B lebih sukar diubah suai. |
| **Kebolehbacaan Kod** | Atur Cara A lebih kemas dan mudah difahami. Fungsi mempunyai nama yang jelas dan spesifik, manakala Atur Cara B memerlukan pembacaan keseluruhan untuk difahami. | Atur Cara A lebih mesra pembangun, terutamanya dalam projek besar atau pembangunan berkumpulan. |
| **Kebolehlanjutan** | Atur Cara A lebih mudah ditambah baik dengan fungsi baharu seperti kiraDiameter() atau kiralsipadu(). Kod Atur Cara B sukar dikembangkan tanpa mengubah kod asal. | Atur Cara A lebih sesuai untuk sistem yang perlu dikembangkan atau dikemaskini dari semasa ke semasa. |

Penerangan Perbandingan – A banding B 1m
Justifikasi -1m

3(b)

| | |
|---|---|
| INPUT | jejariBulatan |
| PROSES | luasBulatan = PI*jejari*jejari |
| | perimeterBulatan = 2*PI*jejari |
| | JIKA (pilihan == 1) maka<br>pergi ke paparLuas(jejari);<br>JIKA TIDAK (pilihan == 2) {<br>pergi ke paparPerimeter(jejari);<br>SELAINNYA<br>papar ("Pilihan tidak sah."); |
| OUTPUT | luasBulatan,perimeterBulatan |

1m
1m
1m
1m
1m+1m

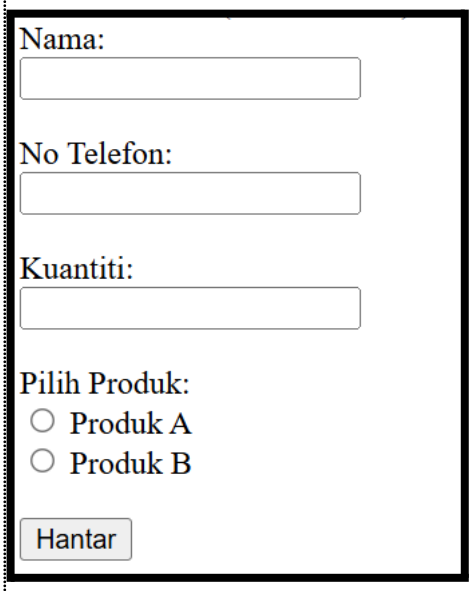
3(c)

| Penambahbaikan | Sebab / Kelebihan |
|---|---|
| 1. **Paparkan menu pilihan dalam bentuk label lebih jelas** | Meningkatkan kefahaman pengguna dengan menunjukkan maksud pilihan 1 dan 2 sebelum input diminta. |
| 2. **Sahkan input sebelum digunakan (Validation)** | Elak ralat atau input pelik seperti nombor negatif untuk jejari dan pilihan tidak wujud. |
| 3. **Pisahkan logik pengiraan dan paparan** | Kaedah paparLuas dan paparPerimeter boleh dipecah dua: kiraLuas() dan papar(), ikut SRP. |
| 4. **Sediakan pilihan ulang semula** | Benarkan pengguna membuat lebih dari satu pengiraan dalam satu sesi — lebih mesra pengguna. |
| 5. **Gunakan switch-case ganti if-else (jika pilihan bertambah)** | Lebih kemas dan tersusun jika ada lebih banyak pilihan dikendalikan (contohnya Luas, Perimeter, Diameter). |
| 6. **Tambahkan komen penjelasan dalam kod** | Membantu pembaca atau pelajar faham maksud setiap bahagian kod dengan lebih mudah. |
| 7. **Format output kepada dua tempat perpuluhan** | Menjadikan output lebih kemas dan profesional untuk nilai bulatan seperti 3.14, 6.28 dan sebagainya. |

1m+1m

Cadangan yang relevan – 1m
Sebab /Kelebihan – 1m

4(a)




Penerangan:

Kotak Nama+Label – 1m

Kotak NoTel +Label – 1m

Kotak Kuantiti+label – 1m

Label Pilih produk+Butang radio – 1m

Butang "Hantar" – 1m

4(b)

```
<form id="orderForm">
 <label>Nama:</label><br>
 <input type="text" id="nama"><br><br>

 <label>No Telefon:</label><br>
 <input type="text" id="telefon"><br><br>

 <label>Kuantiti:</label><br>
 <input type="number" id="kuantiti"><br><br>

 <label>Pilih Produk:</label><br>
 <input type="radio" name="produk" value="Produk A"> Produk A<br>
 <input type="radio" name="produk" value="Produk B"> Produk B<br><br>

 <button type="button" onclick="hantarBorang()">Hantar</button>
</form>
```

**Penerangan**

- **Ada tag form (lengkap) -1m**
- **Ada keterangan label+kotak input -1m (Nama, No telefon, Kuantiti-3M)**
- **Ada pilih produk+butang radio – 1m**
- **Ada button hantarBorang – 1m (nama fungsi boleh apa2 asal bermakna)**

4(c)

```
<SCRIPT>
function hantarBorang() {
 var nama = document.getElementById("nama").value;
 var telefon = document.getElementById("telefon").value;
 var kuantiti = document.getElementById("kuantiti").value;
 var produk = document.querySelector('input[name="produk"]:checked');

 if (nama === "" || telefon === "" || kuantiti === "" || !produk) {
  alert("Sila lengkapkan semua medan!");
```

```
  return;
}

if (telefon.length !== 10 && telefon.length !== 11) {
  alert("Nombor telefon mestilah 10 atau 11 digit.");
  return;
}

var ringkasan = "Nama: " + nama + "\nTelefon: " + telefon +
              "\nKuantiti: " + kuantiti +
              "\nProduk: " + produk.value;
                        alert("Sahkan pesanan anda:\n\n" + ringkasan);
}

</SCRIPT>
```

**Penerangan:**

- **Ada nama fungsi – 1m**
- **Ada isytihar dan terima nilai dari kotak -1m**
- **Ada kotak pop up utk nama -1m**
- **Ada kotak pop up utk nombor telefon – 1m**
- **Ada paparan pengesahan – 1m**