Jawab:

1. **Tahapan-tahapan proses masuk ke dalam memori antara lain:**
* **Tahapan kompilasi**: Source program (source code) dikompilasi menjadi object module (object code)
* **Tahapan link & load**: Object module di-link dengan object module lain menjadi load module (execution code) kemudian di-load ke memori untuk dieskekusi
* **Tahapan eksekusi**: Mungkin juga dilakukan dynamic linking dengan resident library
1. **Yang dimaksud dengan alamat logika dan alamat fisik adalah:**
	* Alamat logika adalah alamat yang di-generate oleh CPU (disebut juga virtual alamat)
		+ Berdasarkan eksekusi program
		+ Note: Besarnya alamat program dapat lebih besar dari kapasitas memori fisik.
	* Alamat fisik adalah alamat yang dikenal oleh unit memori
		+ Alamat sebenarnya yang digunakan untuk mengakses memori.
2. **Pengertian dan kegunaan dari Pemanggilan dinamis adalah:**

Seluruh proses dan data berada di memori fisik ketika dieksekusi. Ukuran dari memori fisik terbatas. Untuk mendapatkan penggunaan ruang memori yang baik, kita melakukan pemanggilan secara dinamis. Dengan pemanggilan dinamis, sebuah rutin tidak akan dipanggil jika tidak diperlukan. Semua rutin diletakkan dalam disk dengan format yang dapat dialokasikan ulang. Progam utama ditempatkan dalam memori dan dieksekusi. Jika sebuah rutin memanggil rutin lainnya, maka akan diperiksa terlebih dahulu apakah rutin tersebut ada di dalam memori atau tidak, jika tidak ada maka linkage loader akan dipanggil untuk menempatkan rutin-rutin yang diinginkan ke memori dan memperbaharui tabel alamat program untuk menyesuaikan perubahan. Kemudian kendali diberikan pada rutin yang baru dipanggil tersebut.

Keuntungan dari pemanggilan dinamis adalah rutin yang tidak digunakan tidak pernah dipanggil. Metode ini berguna untuk kode dalam jumlah banyak, ketika muncul kasus-kasus yang tidak lazim, seperti rutin yang salah. Dalam kode yang besar, walaupun ukuran kode besar, tapi yang dipanggil dapat jauh lebih kecil.

Pemanggilan Dinamis tidak memerlukan bantuan sistem operasi. Ini adalah tanggung jawab para pengguna untuk merancang program yang mengambil keuntungan dari metode ini. Sistem operasi dapat membantu pembuat program dengan menyediakan kumpulan data rutin untuk mengimplementasi pemanggilan dinamis.

1. **Kegunaan dari overlay adalah:**

Teknik Overlays biasanya digunakan untuk memungkinkan sebuah proses mempunyai jumlah yang lebih besar dari memori fisik daripada alokasi memori yang diperuntukkan. Tekniknya adalah menyimpan di memori hanya instruksi dan data yang diperlukan pada satu waktu. Jika intruksi lain diperlukan, maka instruksi tersebut diletakkan di ruang memori menggantikan instruksi yang tidak digunakan lagi. Overlay tidak membutuhkan dukungan khusus dari sistem operasi. User dapat mengimplementasikannya secara lengkap menggunakan struktur file sederhana, membaca dari file ke memori dan meloncat ke memori dan mengeksekusi instruksi read yang lebih baru. Sistem operasi memberitahu hanya jika terdapat I/O yang melebihi biasanya. Penggunaan overlay terbatas untuk beberapa sistem yang mempunyai jumlah memori fisik terbatas dan kekurangan dukungan H/W untuk teknik yang lebih lanjut.