

# 2020.11.9 分布式系统期中考试

- (1) 分散システムはそのユーザに対して単一の  したシステムとして見える。  
(A)矛盾 (B)分裂 (C)首尾一貫
- (2)  分散では、システムを役割が異なる構成要素に分割する。  
(A)垂直 (B)水平 (C)鉛直
- (3) よく知られたサービスについて割り当てられたポートを  ポートと呼ぶ。  
(A)ステートフル (B)ウェルノウン (C)TCP
- (4) Fuggetta のコード移動のモデルにおいて、弱移動性とは、 セグメントのみを移動できることを意味する。  
(A)リソース (B)実行 (C)コード
- (5) RPC による遠隔通信において、パラメータを 1 つのメッセージに包み込むことをパラメータ  という。  
(A)スタブ (B)コール (C)マーシャリング
- (6)  RPC では、サーバの応答が帰ってくるまで、クライアントは次の処理に進むことができない。  
(A)同期型 (B)非同期型 (C)保留同期型
- (7)  転送モードでは、一定の最大遅延が定められ、データはそれより早く到着していれば問題ない。  
(A)同期型 (B)非同期型 (C)等時性
- (8) NFS URL を構成する 3 つの要素は、プロトコル、リモートマシンの名前、リモートマシンでの  である。  
(A)ユーザ名 (B)パスワード (C)パス
- (9) DNS を構成する名前サーバのうち、 層のものは、組織の部課単位で管理され、特に速やかに応答することが求められる。  
(A)グローバル (B)マネージャ (C)部門管理
- (10) 国際原子時間は、 時計のティック数に基づいて決められる。  
(A)カリウム (B)セシウム (C)マグネシウム

得 点	
--------	--

二、それぞれの説明に最もあてはまる語句を、選択肢から選び、A～Dの記号で答えよ。

- (11) システムのサービスを利用したいときに利用できるという性質。  
☒ (A) 可用性 (B) 一貫性 (C) 局所性 (D) スケーラビリティ
- (12) ユーザが遠くのリソースにアクセスできるという性質。  
 (A) 可用性 (B) 一貫性 (C) 局所性 ☒ (D) アクセス可能性
- (13) システムの規模を拡大しやすいという性質。  
 (A) 可用性 (B) 一貫性 (C) 局所性 ☒ (D) スケーラビリティ
- (14) ACID 特性のうち、並行するトランザクションが互いに干渉しないという性質。  
 (A) 一貫性 (B) 原子性 ☒ (C) 独立性 (D) 永続性
- (15) ACID 特性のうち、1つのトランザクションは外から見て分割できないという性質。  
 (A) 一貫性 ☒ (B) 原子性 (C) 独立性 (D) 永続性
- (16) ACID 特性のうち、コミットされたトランザクションによる変更は永久であるという性質。  
 (A) 一貫性 (B) 原子性 (C) 独立性 ☒ (D) 永続性
- (17) システムの利用者に、リソースの位置を意識させないという性質。  
☒ (A) 位置透過性 (B) 複製透過性 (C) 並行透過性 (D) アクセス透過性
- (18) システムの利用者に、リソースが移動したことを意識させないという性質。  
 (A) 並行透過性 (B) 複製透過性 ☒ (C) 移動透過性 (D) アクセス透過性
- (19) システムの利用者に、リソースのコピーの存在を意識させないという性質。  
 (A) 位置透過性 ☒ (B) 複製透過性 (C) 障害透過性 (D) 並行透過性
- (20) アーキテクチャの一種で、コンポーネントがオブジェクトとして実現され、互いにメソッド呼び出しにより連携する。  
☒ (A) オブジェクトベースアーキテクチャ (B) データ中心型アーキテクチャ  
 (C) 階層型アーキテクチャ (D) イベントベースアーキテクチャ
- (21) アーキテクチャの一種で、複数のコンポーネントが共有のデータ空間を持ち、データの一貫性を保ったままデータの更新や取得を行うことができる。  
 (A) オブジェクトベースアーキテクチャ ☒ (B) データ中心型アーキテクチャ  
 (C) 階層型アーキテクチャ (D) イベントベースアーキテクチャ
- (C) 階層型アーキテクチャ (D) イベントベースアーキテクチャ
- (21) アーキテクチャの一種で、複数のコンポーネントが共有のデータ空間を持ち、データの一貫性を保ったままデータの更新や取得を行うことができる。  
 (A) オブジェクトベースアーキテクチャ ☒ (B) データ中心型アーキテクチャ  
 (C) 階層型アーキテクチャ (D) イベントベースアーキテクチャ
- (22) プロセス内部の実行の筋。並列的に実行することも可能。  
☒ (A) スレッド (B) プログラム (C) コード (D) ディスパッチャ
- (23) メッセージキューイングシステムの基本インタフェースのうち、キューのメッセージを取得するもの（ただし、キューが空だとブロックされるもの）。  
 (A) put ☒ (B) get (C) notify (D) poll
- (24) マルチキャスト通信等において、実際のネットワークの上に作る仮想的なネットワーク。  
 (A) LAN (B) WAN  
 (C) インターネット ☒ (D) オーバレイネットワーク
- (25) 名前解決の手法のうち、エンティティが移動するごとに、位置参照のためのポインタを生成するもの。  
 (A) ブロードキャスト (B) マルチキャスト  
☒ (C) 転送ポインタ (D) ホームベースアプローチ

(26) 主語、目的語、述語の組でリソース間の関連を XML 形式で記述する、リソース記述のためのフレームワーク。

(A)LDAP (B)HTML (C)RDF (D)SGML

(27) 複数のプロセスが互いの終了を待ち、いずれも先に進めなくなる状態。

(A)飢餓 (B)枯渇 (C)空腹 (D)デッドロック

- × (28) 分散システムでは、システムを構成する計算機のうち 1 台が故障すると、必ずサービス停止に陥る。
- (29) 分散システムでは、システムを構成する計算機の数を増やすことで、全体としての性能を向上させることができる。
- (30) 分散システムでは、利用者はシステムが多数の計算機で構成されていることを意識しなくてもよい。
- × (31) 分散システムにおける開放性とは、パスワードを設けない等によって誰でも利用可能であるという性質である。
- × (32) 分散システムを構成する計算機の種類は一樣である。
- × (33) クライアントサーバシステムでは、一台の計算機は必ずクライアントかサーバのどちらかだけの役割を持つ。
- (34) ピアツーピアシステムでは、原則として各ノードが対等の役割を持つ。
- × (35) シンククライアントでは、ファットクライアントと比べて、クライアントが持つ役割が大きい。
- × (36) 並行サーバでは、クライアントからの要求を処理している間は次の要求を受け付けられない。
- (37) 一般的に、新規プロセスの生成は新規スレッドの生成より高コストである。
- (38) 分散システムにおけるコード移動の利点として、処理待ちの減少が挙げられる。
- × (39) コネクション型のプロトコルでは、データ交換前に明示的なコネクション確立は不要である。
- × (40) プロセス間の一時通信の特徴として、受信側はメッセージ投入時に実行されている必要がないことが挙げられる。
- × (41) あるエンティティに付けられた名前は、他のエンティティに再利用される可能性はない。
- (42) NFS により、ネットワークの先にある他のマシンのファイルシステムをマウントできる。
- (43) DNS の反復名前解決は、再帰名前解決よりもサーバの性能が比較的低くても良いという利点がある。
- (44) 国際原子時間に閏秒を導入するなどの調整を加え、世界的に使用されている基準時刻を協定世界時と呼ぶ。
- × (45) 最大ドリフト率  $\rho$  の 2 台のマシン間で時間のズレを  $\delta$  以内に抑えたい場合に最低限必要な再同期の周期は  $\delta / \rho$  である。
- (46) NTP では、階層-2 サーバは階層-1 サーバの時刻に合わせる。
- × (47) Berkeley アルゴリズムによるクロック同期では、調整後の時刻と現実の時刻が必ず一致する。