PROFESSIONAL CAREER CLUB

FREE CERTIFICATION PREP

FORMULA SHEET

CPA ISC

Information Systems & Controls · Discipline section

37 formulas **3** subject areas

HOW TO USE THIS SHEET

ISC is one of three discipline sections — you sit ONE of BAR, ISC or TCP. ISC is the newest, the least quantitative, and the least understood, which means the least reliable study material.

A warning about expectations: ISC is NOT an IT exam. It is an ASSURANCE exam that happens to be about IT systems. You are not being asked to configure a firewall — you are being asked what could go wrong, what control addresses it, and what evidence would satisfy an auditor. Keep the auditor's seat in mind and the whole section makes sense.

There are only three real formulas in ISC (risk, ALE and availability). Everything else is frameworks, definitions and distinctions — so read this sheet for the boundaries, not the arithmetic.

The ▸ notes flag the exam-relevant angle: what the control is FOR, where it fails, and what an auditor would ask about it.

INFORMATION SYSTEMS & DATA MANAGEMENT · 15 formulas

Architecture, resilience and IT general controls. The auditor's view of how systems are built and kept running.

IT General Controls — the four domains

```
(1) Logical Access – authentication, authorisation, segregation of duties (2) Change Management – request, approve, test, deploy (3) IT Operations – job scheduling, backups, incident management, monitoring (4) Systems Development / Acquisition – SDLC, vendor management
```

Where: ITGCs support reliance on application-level controls.

▸ *This is the most important concept in ISC. If ITGCs FAIL, the auditor cannot rely on ANY automated application control — because a broken change-management process means the control you tested in March might be different in September.*

Application controls — input, processing, output

```
INPUT: edit checks, validation rules, batch totals, completeness checks
PROCESSING: run-to-run totals, control totals, logic checks, exception reports
OUTPUT: distribution restrictions, reconciliation, error-log review
```

Where: Transaction-level controls, sitting on top of ITGCs.

▸ *Application controls are only as trustworthy as the ITGCs beneath them. A perfect input validation rule is worthless if any developer can change the code without approval.*

RPO vs RTO

```
RPO (Recovery Point Objective) – the maximum acceptable DATA LOSS, measured in time. How old can the restored data be?
RTO (Recovery Time Objective) – the maximum acceptable DOWNTIME.
```

Where: Two different objectives.

▸ *RPO drives BACKUP FREQUENCY. RTO drives RECOVERY INFRASTRUCTURE. An RPO of five minutes demands continuous replication; an RTO of five minutes demands a hot site. They are independent — and the exam will give you one and ask for the other.*

Business Impact Analysis — required outputs

```
(1) Critical processes and their dependencies (2) MTD (maximum tolerable downtime) (3) RTO (4) RPO (5) Resource requirements (6) Financial and operational impact
```

Where: Constraint: RTO ≤ MTD.

▸ *The BIA comes FIRST — it tells you what to protect and how fast. Building a disaster recovery plan without a BIA means protecting the wrong things at the wrong cost.*

Disaster recovery site tiers

```
HOT: fully equipped with live data. RTO under an hour. Highest cost.
WARM: hardware and network ready, no current data. RTO hours to a day. Mid cost.
COLD: facility and utilities only. RTO days to weeks. Lowest cost.
```

Where: Cost trades directly against recovery speed.

▸ *The choice is dictated by the RTO from the BIA, not by preference. If the business can tolerate three days down, a hot site is a waste of money — and an auditor should say so.*

MTBF and system availability

```
MTBF = Total operational time / Number of failures
Availability = MTBF / (MTBF + MTTR)
```

Where: MTTR = mean time to repair.

▸ *Two ways to raise availability: fail less often (higher MTBF) or recover faster (lower MTTR). Often the second is far cheaper than the first — which is a genuinely useful engineering insight.*

Data classification levels

```
Government: Top Secret → Secret → Confidential → Unclassified
Commercial: Restricted → Confidential → Internal → Public
```

Where: Drives access controls, handling and retention.

▸ *Classification comes FIRST. You cannot protect data appropriately until you know what it is worth — which is why classification is the opening step of the data lifecycle.*

Data lifecycle (CRUD-DD)

```
Create (classify) → Read (access controls) → Update (audit trails) → Delete (logical removal) → Disclose (DLP and authorisation) → Destroy (overwrite, degauss, or physical destruction)
```

Where: Six stages, each with its own control.

▸ *DELETE and DESTROY are different. Deletion is logical — the data is still recoverable. Destruction is physical and irreversible. Selling a decommissioned server that was only "deleted" is a genuine breach.*

Database normalisation — 1NF, 2NF, 3NF

```
1NF: atomic values, no repeating groups
2NF: 1NF + no PARTIAL dependencies (non-key attributes depend on the FULL primary key)
3NF: 2NF + no TRANSITIVE dependencies (non-key attributes depend only on the key, not on each other)
```

Where: Each form builds on the previous one.

▸ *Normalisation reduces redundancy, which reduces the chance of INCONSISTENT data — the same fact stored twice can disagree with itself. That data-integrity argument is why an auditor cares at all.*

ETL vs ELT, warehouse vs lake

```
ETL: Extract → Transform → Load (transform BEFORE loading; feeds a structured warehouse)
ELT: Extract → Load → Transform (load raw, transform later; feeds a lake)
Data warehouse: structured, schema-ON-WRITE, analytics-optimised
Data lake: raw and unstructured, schema-ON-READ, cheap storage
```

Where: Where the transformation happens is the whole difference.

▸ *Schema-on-write forces structure up front (governed, slower). Schema-on-read defers it (flexible, and prone to becoming a "data swamp" with no governance). That governance risk is the audit angle.*

Cloud service models

```
IaaS: the customer manages OS + runtime + applications + data (AWS EC2)
PaaS: the customer manages applications + data only (Heroku)
SaaS: the customer manages data and configuration only (Salesforce)
```

Where: The provider manages everything BELOW the customer's layer.

▸ *This is the SHARED RESPONSIBILITY MODEL, and it is the heart of cloud auditing. Moving to SaaS does not outsource your accountability — it just means you now need a SOC report to cover what you no longer control.*

Cloud deployment models

```
Public (shared, multi-tenant) · Private (dedicated to one organisation) · Community (shared by organisations with common concerns) · Hybrid (public + private, orchestrated together)
```

Where: Who else is on the infrastructure?

▸ *The audit concern with PUBLIC cloud is multi-tenancy — your data shares hardware with strangers. Logical separation is the control, and a SOC 2 is how you get comfort that it works.*

SDLC — the seven phases

```
1. Planning / feasibility  2. Requirements analysis  3. System design  4. Development / coding  5. Testing / QA  6. Implementation / deployment  7. Maintenance
```

Where: ITGC change management covers phases 3–6.

▸ *Key risks by phase: scope creep in requirements (2), inadequate testing (5), poor documentation (any). The classic control failure is a developer with access to PRODUCTION — a segregation-of-duties breach that lets code bypass testing entirely.*

Development methodologies

```
WATERFALL: sequential phases, heavy documentation, rigid — suits stable requirements
AGILE (Scrum/Kanban): iterative sprints, flexible — suits changing requirements
DevOps: development and operations integrated, CI/CD, automated testing and deployment
```

Where: Three different approaches to the same lifecycle.

▸ *Agile and DevOps do NOT remove the need for controls — they AUTOMATE them. In a CI/CD pipeline, the approval gates and automated tests ARE the change-management control, and that is what an auditor tests.*

Backup site tiers

```
Cold: basic infrastructure only — lowest cost, longest RTO (days to weeks)
Warm: partial equipment — moderate cost and RTO (hours to days)
Hot: fully mirrored, real-time — highest cost, lowest RTO (minutes to hours)
```

Where: The same cost/speed trade-off as DR sites.

▸ *The backup itself is not the control — the RESTORE is. An organisation that backs up religiously and never tests a restore has no recovery capability, only the illusion of one. Auditors ask when the restore was last tested.*

EXAM TIPS — INFORMATION SYSTEMS & DATA MANAGEMENT

- If ITGCs fail, no automated application control can be relied upon. That single dependency is the organising idea of the whole section.
- RPO = data loss (drives backup frequency). RTO = downtime (drives infrastructure). Keep them apart; the exam deliberately mixes them.
- Cloud does not transfer accountability. It transfers the CONTROL, which is why you then need a SOC report to cover it.

SECURITY, CONFIDENTIALITY & PRIVACY · 16 formulas

Threats, cryptography, frameworks and privacy law. The largest area, and almost entirely conceptual.

The CIA triad

```
Confidentiality (only authorised parties see data — threat: theft) · Integrity (data accurate and unaltered — threat: malicious modification) · Availability (systems usable when needed — threat: denial of service)
```

Where: The three security objectives.

▸ *Map every control to one pillar. Encryption serves confidentiality; hashing serves integrity; redundancy serves availability. When a question describes a control, ask which pillar it protects and the answer usually falls out.*

Information security risk

```
Risk = Threat × Vulnerability × Impact
```

Where: Controls reduce vulnerability or impact — not the threat.

▸ *You cannot eliminate the THREAT (hackers exist regardless of what you do). You reduce the VULNERABILITY (patch it) or the IMPACT (encrypt it, so the stolen data is useless). What remains after controls is RESIDUAL risk.*

Annual loss expectancy

```
ALE = SLE × ARO
SLE = single loss expectancy (loss per incident)
ARO = annualised rate of occurrence
```

Where: The expected annual cost of a risk.

▸ *A control is cost-effective only if its cost is LESS than the reduction in ALE it achieves. Spending $500,000 to prevent a $50,000 annual expected loss is a control failure of a different kind — and that judgement is exactly what the exam wants.*

Cryptography — symmetric, asymmetric, hashing

```
SYMMETRIC (AES, DES): the SAME key encrypts and decrypts. Fast; the problem is key distribution.
ASYMMETRIC (RSA, ECC): a public/private key PAIR. Slower; it solves key exchange and enables digital signatures.
HASHING (SHA-256): ONE-WAY, fixed length. For integrity and password storage.
```

Where: Three tools, three jobs.

▸ *Hashing is NOT encryption — it cannot be reversed, which is precisely why passwords are stored hashed. In practice, TLS uses asymmetric crypto to exchange a symmetric key, then symmetric crypto for the traffic: you get the security of one and the speed of the other.*

Encryption key sizes

```
AES-128 adequate; AES-256 for sensitive data
RSA-2048 minimum; RSA-4096 for long-term
ECC-256 ≈ RSA-3072 in strength
NIST: 112-bit minimum through 2030; 128-bit beyond
```

Where: Key length drives strength.

▸ *ECC achieves equivalent strength with a much shorter key, which is why it dominates mobile and IoT — less computation, less battery, same security.*

Encryption at rest vs in transit

```
AT REST: full-disk (BitLocker, FileVault), database-level (TDE), file-level — AES-256 standard
IN TRANSIT: TLS 1.2+ (HTTPS), IPsec VPN, S/MIME or PGP for email
```

Where: Deploy BOTH.

▸ *At-rest encryption alone leaves the data exposed during transmission — and in-transit alone leaves it exposed on the disk. They protect different moments in the data's life, and neither substitutes for the other.*

Public Key Infrastructure

```
(1) Certificate Authority – issues and signs certificates  (2) Registration Authority – verifies identity before issuance  (3) X.509 digital certificates  (4) CRL / OCSP for revocation  (5) The public/private key pair
```

Where: Enables TLS, S/MIME and code signing.

▸ *REVOCATION is the weak link. A compromised certificate is only harmless if clients actually check the revocation list — and many do not. That gap between design and practice is a favourite exam theme.*

Authentication factors and MFA

```
Something you KNOW (password, PIN)
Something you HAVE (token, smartcard, push notification)
Something you ARE (biometric – fingerprint, iris, face)
```

Where: MFA = two or more DIFFERENT factors.

▸ *A password plus a security question is NOT multi-factor — both are "something you know". Two factors from the SAME category give no additional protection, and that is the trap.*

NIST password guidelines (SP 800-63B)

```
Minimum length 8 (15 for administrators)
NO composition/complexity rules
NO forced rotation unless there is evidence of compromise
Screen against lists of breached passwords
ALLOW paste and password managers
```

Where: A deliberate reversal of decades of practice.

▸ *Forced rotation and complexity rules make passwords WORSE — users respond with Password1!, Password2!, and sticky notes. NIST now recommends against both. If a question offers "require quarterly password changes" as the best control, that is the wrong answer.*

Defence in depth — seven layers

```
1. Perimeter (firewalls, DMZ)  2. Network (segmentation, VLANs, IDS/IPS)  3. Endpoint (antivirus, EDR)  4. Application (input validation, secure coding)  5. Data (encryption, DLP)  6. Identity (MFA, RBAC)  7. Physical (locks, cards, surveillance)
```

Where: Layered, overlapping controls.

▸ *The premise is that EVERY control will eventually fail. Layering means one failure is not fatal. A single perfect control is a worse design than several imperfect ones.*

Firewall types

```
STATELESS: matches each packet against an ACL. Fast; no session memory.
STATEFUL: tracks connection state in a session table.
WAF: inspects HTTP/HTTPS at the APPLICATION layer (SQL injection, XSS).
NGFW: stateful + deep packet inspection + IDS/IPS + application and user awareness.
```

Where: Increasing sophistication.

▸ *A traditional firewall CANNOT stop SQL injection — the traffic is legitimate HTTP on port 443. You need a WAF, which understands what the request is actually asking the application to do.*

Common attack vectors

```
1. Phishing and social engineering (targets HUMANS)  2. Malware (ransomware, viruses, worms, trojans)  3. SQL injection  4. Cross-site scripting (XSS)  5. DDoS
```

Where: The top five.

▸ *PHISHING is the most common initial vector, and no technical control fully prevents it — the attack targets people, not systems. That is why training is a control, and why an exam answer emphasising user awareness is often correct.*

Data Loss Prevention

```
NETWORK DLP – inspects data in MOTION (email, web)
ENDPOINT DLP – an agent on devices; controls USB, printing, screenshots
STORAGE/CLOUD DLP – discovers and classifies data AT REST
```

Where: Prevents exfiltration of PII, IP and financial data.

▸ *DLP depends entirely on CLASSIFICATION. It can only protect data it can recognise — which is why the data lifecycle begins with classification, not with tooling.*

NIST Cybersecurity Framework (IPDRR)

```
Identify (asset management, risk assessment, governance) · Protect (access control, training, data security) · Detect (anomalies, continuous monitoring) · Respond (planning, communications, mitigation) · Recover (recovery planning, improvements)
```

Where: Outcome-based and VOLUNTARY.

▸ *Five functions. Most organisations over-invest in PROTECT and under-invest in DETECT and RESPOND — which is why breaches go undiscovered for months. That imbalance is a standard exam observation.*

Incident response phases (PICERL)

```
Preparation → Identification → Containment → Eradication → Recovery → Lessons Learned
```

Where: NIST SP 800-61.

▸ *CONTAINMENT comes before ERADICATION — stop the bleeding before you hunt the cause. And LESSONS LEARNED is the phase everyone skips, which is why the same incident recurs. Test the plan with tabletop exercises before you need it.*

Privacy regulations — GDPR, CCPA, HIPAA

```
GDPR (EU): consent; rights of access, deletion and portability; 72-HOUR breach notification; fines up to 4% of GLOBAL revenue
CCPA (California): the right to know, delete, and opt out of sale
HIPAA (US health): PHI Privacy and Security Rules; breach notice within 60 days
```

Where: Three regimes, three notification clocks.

▸ *GDPR's 72 hours is the number they test, and its extraterritorial reach is the concept — it applies to anyone processing EU residents' data, wherever the company sits. That is why it reshaped global privacy practice.*

EXAM TIPS — SECURITY, CONFIDENTIALITY & PRIVACY

- Every control maps to a CIA pillar. Ask which one it protects, and the purpose of the control becomes obvious.
- Risk = Threat × Vulnerability × Impact. Controls reduce vulnerability and impact, never the threat. What is left over is residual risk, and the organisation must accept it explicitly.
- MFA requires two DIFFERENT factor types. Two passwords are not two factors — that is the most common distractor in the section.

SOC ENGAGEMENTS · 6 formulas

The service organisation reports. Small, precise, and highly examinable — a CPA-specific topic that rewards exact vocabulary.

SOC 1 vs SOC 2 vs SOC 3

```
SOC 1: internal control over FINANCIAL REPORTING at the service organisation. Audience: USER AUDITORS. Restricted.
SOC 2: the Trust Services Criteria. Audience: management and customers. Restricted.
SOC 3: the SAME criteria as SOC 2, but a public general-use summary with less detail.
```

Where: Purpose and audience differ.

▸ *SOC 1 exists to serve the USER ENTITY'S FINANCIAL STATEMENT AUDITOR — that is its entire reason for being. SOC 2 serves customers worried about security. If the question is about financial reporting, the answer is SOC 1.*

Type 1 vs Type 2

```
TYPE 1: the DESIGN of controls at a POINT IN TIME. A snapshot. Quicker, less reliable.
TYPE 2: DESIGN plus OPERATING EFFECTIVENESS over a PERIOD (typically 6–12 months). Preferred by user auditors.
```

Where: Both apply to SOC 1 and SOC 2.

▸ *A Type 1 tells you the control was well designed on one day. A Type 2 tells you it actually WORKED for a year. A user auditor seeking to place reliance needs a Type 2 — a Type 1 supports almost no reliance at all.*

Trust Services Criteria — the five categories

```
(1) SECURITY — required in EVERY SOC 2  (2) Availability  (3) Processing Integrity  (4) Confidentiality  (5) Privacy
```

Where: The service organisation selects which categories beyond Security to include.

▸ *SECURITY is the only mandatory one — it is called the "common criteria". A SOC 2 covering only Security is entirely valid, so always check WHICH criteria a report actually covers before relying on it.*

Carve-out vs inclusive method

```
CARVE-OUT: the subservice organisation's controls are EXCLUDED from the report. The user must obtain a separate SOC for the subservice.
INCLUSIVE: the subservice organisation's controls are INCLUDED (it provides a description and cooperates with the auditor).
```

Where: Carve-out is the default and by far the most common.

▸ *Under a carve-out, there is a HOLE in your assurance — and it is your job to fill it by obtaining the subservice organisation's own SOC report. Candidates miss that the carve-out shifts work to the user, rather than eliminating it.*

Bridge (gap) letter

```
A letter from the service organisation covering the period from the SOC report's period-end to the user's current date. It asserts: (1) no material changes to controls (2) no significant incidents or breaches.
```

Where: Maximum gap of roughly six months.

▸ *A bridge letter is MANAGEMENT'S ASSERTION — it is NOT audited and it is NOT a substitute for a new SOC report. It is comfort, not evidence, and an auditor should treat it accordingly.*

SOC report types — summary

```
SOC 1 → ICFR → for financial statement auditors
SOC 2 → Trust Services Criteria → for management and customers
SOC 3 → public summary of a SOC 2
Type 1 = design only (point in time) · Type 2 = design + operating effectiveness (period)
```

Where: Two dimensions: which report, and which type.

▸ *Build a grid: report number (what is being examined) crossed with type (design only, or design and effectiveness). Every SOC question sits in one cell of that grid.*

EXAM TIPS — SOC ENGAGEMENTS

- SOC is a two-dimensional grid: report NUMBER (1, 2, 3) crossed with TYPE (1 or 2). Locate the question in the grid before answering.
- SOC 1 = financial reporting = for the user's auditor. That single association resolves most SOC questions.
- A bridge letter is an unaudited management assertion. It never replaces a SOC report.

PASSING ISC

What ISC actually is

- An ASSURANCE exam about IT — not an IT exam. You are always in the auditor's seat: what could go wrong, what control addresses it, what evidence would satisfy you.
- Almost entirely conceptual. Three formulas (risk, ALE, availability) and hundreds of distinctions.
- The newest section, so study material quality varies more than for the others. Anchor on the AICPA blueprint rather than on third-party summaries.

Choosing ISC as your discipline

- Suits candidates heading into IT audit, internal audit, advisory, cybersecurity or risk — and anyone who would rather learn frameworks than tax law or variance formulas.
- It is the least quantitative of the three disciplines by a wide margin. If numbers are your weakness, this is the strategic choice.
- It overlaps meaningfully with AUD — the control mindset, the evidence hierarchy, the reliance logic all carry across.

The high-yield core

- ITGCs and why their failure destroys reliance on every automated control beneath them.
- SOC reports — the number/type grid, and the carve-out gap.
- RPO vs RTO, and the BIA that produces them.
- The CIA triad, and mapping every control to a pillar.
- The cloud shared responsibility model: IaaS, PaaS, SaaS and what the customer still owns.