Las competencias profesionales que debe adquirir un ingeniero en sistemas computacionales para presentar el examen EGEL son:

**Matemáticas**

1. Aplica la fórmula de combinación y permutación para la solución de problemas estadísticos.
2. Aplica el Teorema de Bayes para el cálculo de la probabilidad dados múltiples eventos.
3. Genera números aleatorios mediante métodos matemáticos.
4. Aplica las tablas de verdad AND, NOT, OR, XOR, XAND, NAND, NOR; en problemas matemáticos mediante soluciones de programación.
5. Calcula la media, mediana, moda, desviación estándar y varianza de problemas estadísticos dados.

**Sistemas Operativos**

1. Crea y administra cuentas FTP con permisos especifico en Linux.
2. Instala un servidor WEB montando una página de ejemplo con acceso remoto, empleando Linux.
3. Distingue las diferencias entre los entornos gráficos más populares para LINUX (GNOME, KDE, UNITY, etc.).
4. Emplea comandos básicos de Linux orientados a la administración de redes (ping, ifconfig, ethtool, fping, dig, fuser, ipcalc, ipcount, host, hostname, ps, kill, iptab, iptables, route, etc).
5. Comprende la arquitectura y funciones de un sistema operativo.
6. Reconoce las funciones específicas que realiza el microprocesador, RAM, ROM y Caché de una computadora.
7. Administra archivos mediante comandos de Linux.
8. Define el tipo y versión de sistema operativo adecuado en función a las características de hardware de una computadora (Windows, Macintosh, Linux, Solaris, etc.). (Se recomienda estudiar el hardware necesario para cada versión actual y 2 versiones anteriores, de los sistemas operativos).

**Diseño de interfaces gráficas.**

1. Emplea patrones de diseño de software.
2. Aplica usabilidad en el diseño de interfaces gráficas.

**Análisis, diseño y gestión de sistemas de información**

1. Aplica ingeniería de requerimientos como primer paso en el desarrollo de sistemas de información.
2. Identifica qué tipo de ciclo de vida de los sistemas de información (RUP, XP, V, Cascada, Incremental, Evolutivo, Win & Win, , etc.) aplica acorde al análisis del problema.
3. Determina la factibilidad de un proyecto empleando un estudio económico, técnico, operativo y financiero.
4. Determina la factibilidad de llevar a cabo un proyecto realizando un análisis costo-beneficio.
5. Realiza análisis FODA para la toma de decisiones.
6. Identifica las competencias profesionales que debe tener un ingeniero en sistemas computacionales para liderar proyectos.
7. Comprende la importancia de conocer los antecedentes, misión, visión y objetivos; de una empresa como primer acción, previo al desarrollo de una solución informática.
8. Emplea escalas LIKERT para la obtención de datos en el análisis de un sistema.
9. Aplica ITIL para mejorar el servicio que ofrece una empresa a sus clientes.
10. Aplica PMBOK para la administración de proyectos.
11. Aplica SWEBOK como recurso para el desarrollo del software.
12. Conoce los requisitos para obtener cada nivel del modelo CMMI.
13. Conoce las novedades existentes del CMMI (SCAMPI, PCMM).
14. Reconoce los beneficios de la metodología PSP Y TSP.
15. Aplica métodos de implementación de software (paralelo, gradual, directo, etc.) de acuerdo a las características de la empresa en la que se .
16. Decide qué tipo de sistema de información debe emplearse según el nivel al que esté dirigido (TPS, OAS, KWS, MIS, DSS, ESS).
17. Emplea SIX SIGMA como metodología para al mejora de procesos.
18. Elabora pruebas de caja negra y blanca al software que se desarrolla.
19. Emplea la metodología SCRUM para el desarrollo de software.
20. Utiliza la métrica de puntos de función para determinar el tamaño del software.
21. Utiliza el modelo COCOMO para determinar costos de proyectos de software.
22. Calcula número de líneas de un código mediante KLOC.
23. Determina la complejidad de un código mediante Big-O a fin de optimizarlo de acuerdo a las características del hardware disponible.
24. Alinea la estrategia con los objetivos del negocio mediante la metodología Balanced ScoreCard.

**Calidad del software.**

1. Reconoce los requisitos para obtener la norma ISO/IEC TR 15504 (Modelo SPICE).
2. Conoce los requisitos y aplica los estándares de calidad del software más populares en el mercado (9000, 9000-3, 9001, 9003, 9004, 9004-2, 9123, 9126, 32000, 19011, 90003:2004, etc.).
3. Emplea métricas de calidad en el desarrollo de software basadas en tamaño y en función.

**Programación**

1. Realiza pruebas de escritorio sin emplear una computadora para determinar el resultado final que devuelve un método recursivo proponiendo valores iniciales como parámetros.
2. Crea métodos recursivos partiendo de un problema donde se haya empleando estructuras de control cíclicas. (Se recomienda hacer algoritmos que den solución a problemas con características cíclicas sin emplear los bucles conocidos en la programación, empleando recursividad).
3. Distingue las diferencias entre herencia e interfaz en la programación orientada a objetos.
4. Comprende la aplicación de una clase abstracta en proyectos.
5. Manipula los métodos de una cadena.
6. Distingue las diferencias entre un arreglo estático y dinámico.
7. Realiza validaciones de entrada por colisión y por posición.
8. Elabora el algoritmo para efectuar procesos de ordenamiento.
9. Reconoce sistemas de información elaborados en el lenguaje de programación C++.
10. Reconoce sistemas de información elaborados en el lenguaje de programación Java.
11. Reconoce sistemas de información elaborados en el lenguaje de programación Basic.
12. Decide estratégica y analíticamente el lenguaje de programación que debe emplearse acorde a un problema dado, considerando: Ruby, Java, Cobol, C, Python, C#, Visual Basic, Javascript, PHP, ASP, Prolog, Smalltalk y Pascal.
13. Distingue la diferencia entre los lenguajes de programación para fines del desarrollo WEB orientados a Servidor y Cliente.
14. Comprende la lógica de programación de PROLOG.
15. Comprende las desventajas de los lenguajes de programación que emplean máquina virtual para traducir el código objeto.
16. Decide qué lenguaje de programación debe emplearse acorde al sistema operativo disponible y el hardware de las computadoras.
17. Crea algoritmos de autómatas para fines diversos.
18. Conoce los componentes de un compilador.
19. Distingue las ventajas y desventajas de cada uno de los paradigmas de la programación más populares.
20. Implementa el modelo de 3 capas en una aplicación WEB.
21. Conoce las características de cada uno de los frameworks más populares en el mercado orientados a la administración de comunidades y de contenidos (Blog, foro, chat, groupware, CMS, portales, faqs y plataformas de e-learning).
22. Domina técnicas para la modificación dinámica de una página WEB.
23. Identifica los caracteres especiales y secuencias de escape de Java (\\, \\*, \n, etc.)
24. Utiliza tecnologías populares para la creación y administración de documentos de esquema (XML, XSD, DDT, XHTML).
25. Reconoce las características y funciones del recolector de basura.
26. Distingue las diferencias entre los modificadores de los métodos y atributos de una clase.
27. Construye sistemas de información mediante MVC.
28. Emplea bibliotecas y tuplas para almacenar datos en la programación de sistemas de cómputo.
29. Práctica diversas notaciones de codificación al programar (Camel case, Pascal case, Snake case, húngaro, etc.)

**Bases de datos**

1. Reconoce errores de sintaxis en una instrucción SQL avanzada.
2. Distingue la diferencia entre el diagrama relacional y el diagrama entidad-relación.
3. Diseña bases de datos normalizadas
4. Determina la(s) clave(s) primaria(s) de una tabla.
5. Distingue la diferencia entre INNER JOIN, LEFT JOIN Y RIGHT JOIN; al relacionar tablas de una base de datos.
6. Une consultas mediante UNION.
7. Crea consultas dinámicas (TRANSFORM) mediante SQL.
8. Crea TRIGGERS mediante SQL para desencadenar funciones específicas al insertar, eliminar o modificar registros en otras tablas.
9. Crea PROCEDURES y FUNCTIONS mediante SQL para fines de consulta y manipulación de datos.
10. Administra usuarios y privilegios de acceso y uso de una base de datos mediante SQL (CREATE USER, GRANT).
11. Restaura y respalda una base de datos mediante comandos SQL.
12. Manipula variables reservadas de diversos gestores de bases de datos.
13. Programa rutinas empleando estructuras de control mediante instrucciones SQL.
14. Añade INDEX a campos en tablas de acuerdo a las necesidades del sistema de información.
15. Clasifica gestores de bases de datos en función a costos (libres y comerciales).

**UML**

1. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de clases.
2. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de secuencia.
3. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de estados.
4. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de actividades.
5. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de comunicación.
6. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de paquetes.
7. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de objetos.
8. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de despliegue.
9. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de componentes.
10. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de casos de uso.
11. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de colaboraciones.
12. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de visión global de iteración.
13. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de tiempos.
14. Diseña, comprende las características, identifica errores y reconoce la simbología del diagrama de estructura compuesta.
15. Identifica el diagrama que debe emplearse en la fase de Requisitos del ciclo de vida de un sistema de información (casos de uso, actividades y secuencia).
16. Identifica el diagrama que debe emplearse en la fase de Análisis del ciclo de vida de un sistema de información (clases, objetos, comunicación, secuencia, visión global de iteración y estados).
17. Identifica el diagrama que debe emplearse en la fase de Diseño del ciclo de vida de un sistema de información (paquetes, componentes, despliegue, estados, secuencia, comunicación, clases, objetos, colaboraciones, tiempos y estructura compuesta).
18. Identifica el diagrama que debe emplearse en la fase de Construcción del ciclo de vida de un sistema de información (clases, paquetes y secuencia).
19. Identifica el diagrama que debe emplearse en la fase de Pruebas del ciclo de vida de un sistema de información (actividades, clases y estado).
20. Identifica el diagrama que debe emplearse en la fase de Despliegue del ciclo de vida de un sistema de información (diagrama de despliegue).

**Redes**

1. Identifica los protocolos de red por cada capa del Modelo OSI y TCP
2. Determina el dominio de colisión y de broadcast de una red.
3. Comprende el uso de DNS.
4. Optimiza redes basado en un problema en particular mediante hardware.
5. Configura redes inalámbricas.
6. Determina el ISP acorde a un problema dado.
7. Implementa recursos de seguridad informática en una red.
8. Conoce el funcionamiento de la fibra óptica.
9. Conoce la función de los puertos más populares (20-ftp data, 21-ftp, 22-ssh, 23-telnet, 25-smtp, 42-nameserver, 43-nickname, 80-http, 88-kerberos, 101-hostname, 109-pop2, 110-pop3, 143-imap, 213-ipx, 443-https, 995-pop3s).
10. Identifica la función de cada capa de del Modelo OSI.
11. Determina la zona desmilitarizada de una red.
12. Identifica el funcionamiento del DHCP.
13. Identifica el funcionamiento de una VLAN.
14. Configura la IP de una máquina en diferentes sistemas operativos.
15. Instala una red con acceso a Internet mediante comandos de CISCO.
16. Configura un Router en una red mediante comandos de CISCO.
17. Configura una VPN mediante comandos de CISCO.
18. Configura una IPv6 mediante comandos de CISCO.
19. Configura interfaces pasivas mediante comandos de CISCO.
20. Configura máscaras de subred y VLSM (máscaras de subred de tamaño variable) mediante comandos de CISCO.
21. Configura redes para el encaminamiento jerárquico y el intercambio de información mediante los protocolos OSPF, RIP, RIPv2 a través de comandos de CISCO.
22. Configura redes mediante el protocolo EIGRP a través de comandos de CISCO.
23. Reconoce los tipos de cableado de red Ethernet (Cruzado y directo) que se debe emplear dependiendo de los dispositivos de comunicación disponibles.
24. Identifica las características y diferencias entre las topologías de red más populares.

NÚMERO DE COMPETENCIAS PROFESIONALES A DESARROLLAR: 130