Técnicas Básicas para el Desarrollo de Programas de Computadoras Métodos usados por el analista para obtener nformación útil antes de desarrollar un sistema. Pueden ser datos primarios (recogidos directamente) o secundarios (obtenidos y procesados por otros previamente). Entrevista Encuesta Conversación dirigida con propósito Conjunto de preguntas normalizadas a Observación específico. Cuestionario una muestra representativa o a toda la Observar fenómenos, hechos o casos Uso de preguntas y respuestas • Lista estructurada de preguntas para población. en su entorno natural. predefinidas o semiestructuradas. recopilar datos. Objetivo: conocer opiniones, Registrar datos para posterior análisis Busca obtener opiniones y Se diseña en función de los objetivos características o hechos concretos • Es la base de muchos descubrimiento sentimientos del entrevistado Puede ser presencial, telefónica, en de la investigación. científicos. Útil para comprender procesos y Debe evitar ambigüedades y línea o impresa. Ventaja: información directa y real. necesidades internas de la preguntas irrelevantes. Ventaja: obtener datos cuantitativos Desventaja: puede ser subjetiva si no organización. representativos. Es pieza clave para alcanzar los fines se estandariza el procedimiento. Importante: el entrevistado suele Desventaja: riesgo de sesgo si la del estudio. conocer mejor la operación que el muestra no es correcta. analista. Otras Técnicas de Diseño Enfoque Monolítico Programación Programación Modular Programación Convencional Procedimental rogramación Estructurada División del programa en módulos cor Basada en funciones o procedimientos que • Introducida por Dijkstra (años 70). agrupan secuencias repetidas. tareas concretas. Características: • Basada en estructuras de control Código sin estructura. básicas: Dependencia total del programador original Características de un módulo: Secuencial. Ventaias: para mantenimiento. o Alternativa (if, if-else, switch). • Nombre único. • Reduce redundancia. o Repetitiva (while, do-while, for). • Comunicación por parámetros. • Facilita el mantenimiento Uso de descomposición funcional • Puede compilarse y almacenarse po organización. diagramas en árbol. separado. · Adecuada para grandes proyectos. Problemas: • Lenguajes comunes: Pascal, C, Ada. Contiene cabecera (nombre y datos) Difícil depuración y localización de errores. • Métodos: Jackson, Bertini, Lassy, cuerpo (código). Modificaciones costosas. • Tamaño ideal: 20–200 líneas de código. Warnier, Gordfolh. Baja transportabilidad entre sistemas. • Una tarea única y completa. • Desventajas: Redundancia de código. · Difícil determinar el punto óptimo par Falta de documentación. encapsular en funciones. Ventajas: Ejemplo: función que recibe un número • Trabajo en equipo. Enfoque E-P-S (Entrada – Proceso – Salida) Programación Orientada a Objetos (POO) y muestra el anterior, el mismo y e Línea Recta Menos redundancia. siguiente. Mantenimiento más sencillo. • Lenguajes independientes de la máquina, • Entrada: fáciles de aprender. • Paradigma actual dominante. o Datos desde usuario, archivos • Ejemplos históricos: FORTRAN, COBOL, BASIC. · Basada en objetos con atributos y bases de datos. Clasificación: métodos. o Puede ser en línea, fuera de línea o o Imperativos (C, Pascal). Ventajas: Programación Procedimental Programación por lotes. Declarativos (Prolog, Lisp). Encapsulación. Procedimental Proceso: o Orientados a Objetos (C++, Smalltalk). Reutilización. o Diseño del modelo + algoritmo de o Orientados al Problema (SQL, NATURAL). Facilidad de mantenimiento. resolución o Naturales (lenguaje cercano al humano). Método Descendente (Top-Down) Ejemplos: Java (100% POO), C+o Representación: pseudocódigo, Por generación: létodo Ascendente (Bottom-Up) • Se diseña primero el módulo principal (híhrido) diagrama de flujo, diagrama N-S. Se comienza con módulos de bajo nive 1ª: máguina y ensamblador. Objetos interactúan entre sí, cada luego los específicos. Salida: o 2ª: FORTRAN, COBOL. y se avanza al superior. uno con su propio ciclo de vida. · Organización jerárquica. o En pantalla, papel, disco o gráficos. La estructura visual es similar a ∘ 3ª: Pascal. C. • Único módulo con jerarquía o Debe cumplir formato y frecuencia 4ª: SQL, bases de datos. descendente, pero cambia el orden de (principal). definidos o 5ª: IA, procesamiento de lenguaje natural. programación. Se crean diagramas de diseño

estructurado que muestran orden

control y responsabilidad.