Facultad de Ingeniería Ingeniería y de Sistemas

Algoritmia y Estructura de Datos: Introducción

Ing. Juan Carlos Sotelo, M.Sc.

Revisión del Sílabo de Algoritmia y Estructura de Datos

- Desarrollo de algoritmos y aplicación de estructuras de datos para resolver problemas
- Aplicación de un lenguaje de programación
- Aplicación de estructuras de control secuenciales, selectivas y repetitivas

Sumilla

- Operaciones con estructuras de datos estáticas: arreglos, registros, cadenas y sus combinaciones
- Desarrollo de algoritmos de ordenamiento y búsqueda
- Aplicación de subprogramas y recursividad
- Concepto de puntero y operaciones con estructuras de datos dinámicas: listas, pilas, colas y árboles

Revisión del Sílabo de Algoritmia y Estructura de Datos

Competencias

1. Desarrollar el razonamiento lógico para la solución de problemas
2. Seleccionar la estructura de datos más adecuada para representar los datos del problema y facilitar la lógica de solución
3. Integrar diversas estructuras de datos en la solución de un problema
4. Desarrollar la capacidad de análisis y diseño para implementar soluciones basadas en algoritmos.
5. Dominar un lenguaje de programación a nivel intermedio

Revisión del Sílabo de Algoritmia y Estructura de Datos

Evaluación

- EP Examen Parcial (Peso 1)
- EF Examen Final (Peso 2)
- PC1, PC2, PC3 y PC4 Prácticas Calificadas
- PP Promedio de Prácticas
- PF Promedio Final

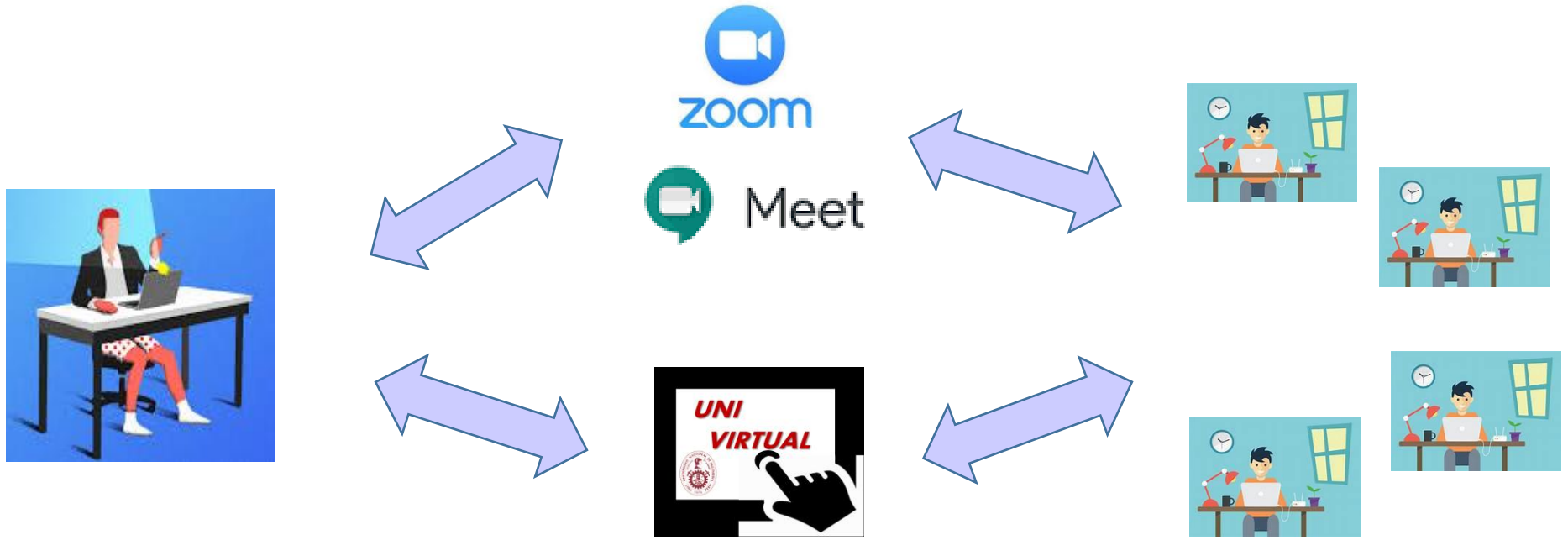
PP = (PC1+PC2+PC3+PC4-mínimo (PC1..PC4)/3)

PF = (EP + EF*2+ PP)/4

Nota: los días de prácticas calificadas y exámenes serán los martes de 8 a 10 am
las prácticas calificadas y exámenes serán únicos para todas las secciones

Modalidad no presencial

- Clases en línea, vía Zoom (martes 8-10 am), vía Google Meet (miércoles de 6-8 pm)
- Evaluaciones via UNIVirtual

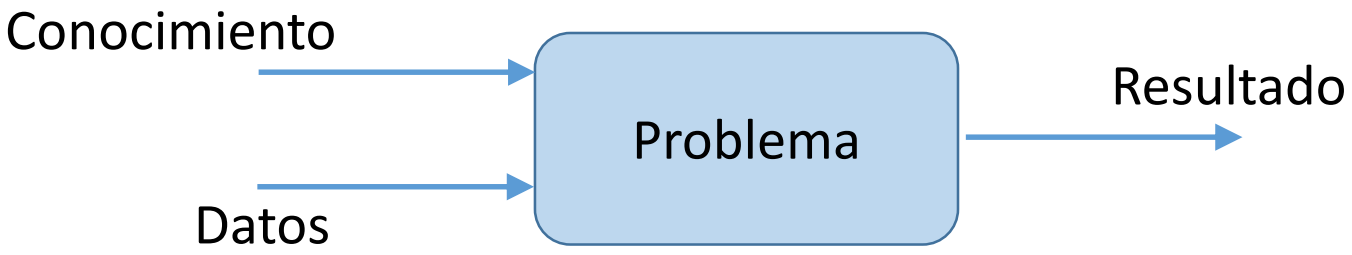


Razones por las que es importante la algoritmia en la formación del ingeniero

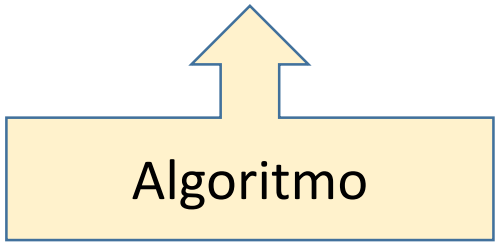
Primera → Desarrolla la capacidad de razonamiento lógico para la solución de problemas

Segunda → Desarrolla la capacidad de especificar de manera precisa

Problema y Algoritmo



Procedimiento seguido
para hallar el resultado



Problema y Algoritmo

Problema

Hallar el Máximo Común Divisor de 2 números enteros positivos

Conocimiento

```
Regla 1: MCD(x,x) = x

Regla 2: si x > y  entonces  MCD(x,y) = MCD(x-y,y)
         si y > x  entonces  MCD(x,y) = MCD(x,y-x)
```

Datos

Un par de números enteros positivos

Problema algorítmico

Especificar la secuencia de pasos que con el conocimiento dado permite hallar el resultado del problema

Problema y Algoritmo

Algoritmo primera aproximación

```
declarar variables  x,y  como enteros

leer x,y
mientras x ≠ y
  aplicar la regla 2
finmientras
escribir el resultado aplicando la regla 1
```

Algoritmo segunda aproximación

```
declarar variables  x,y  como enteros

leer x,y
mientras x ≠ y
  si x > y entonces x = x-y
  en caso contrario y = y-x
finmientras
escribir "Máximo Común Divisor es ", x
```

Algoritmo y código en C++

Algoritmo en pseudocódigo

```
declarar variables  x,y  como enteros

leer x,y
mientras x ≠ y
  si x > y entonces x = x-y
  en caso contrario y = y-x
finmientras
escribir "Máximo Común Divisor es ", x
```

Código e C++