

Análisis de Algoritmos y Estructura de Datos

Laboratorio Complejidad de algoritmos

Prof. Violeta Chang C

Semestre 2 – 2023



Algoritmos y pseudocódigo

- **Contenidos:**

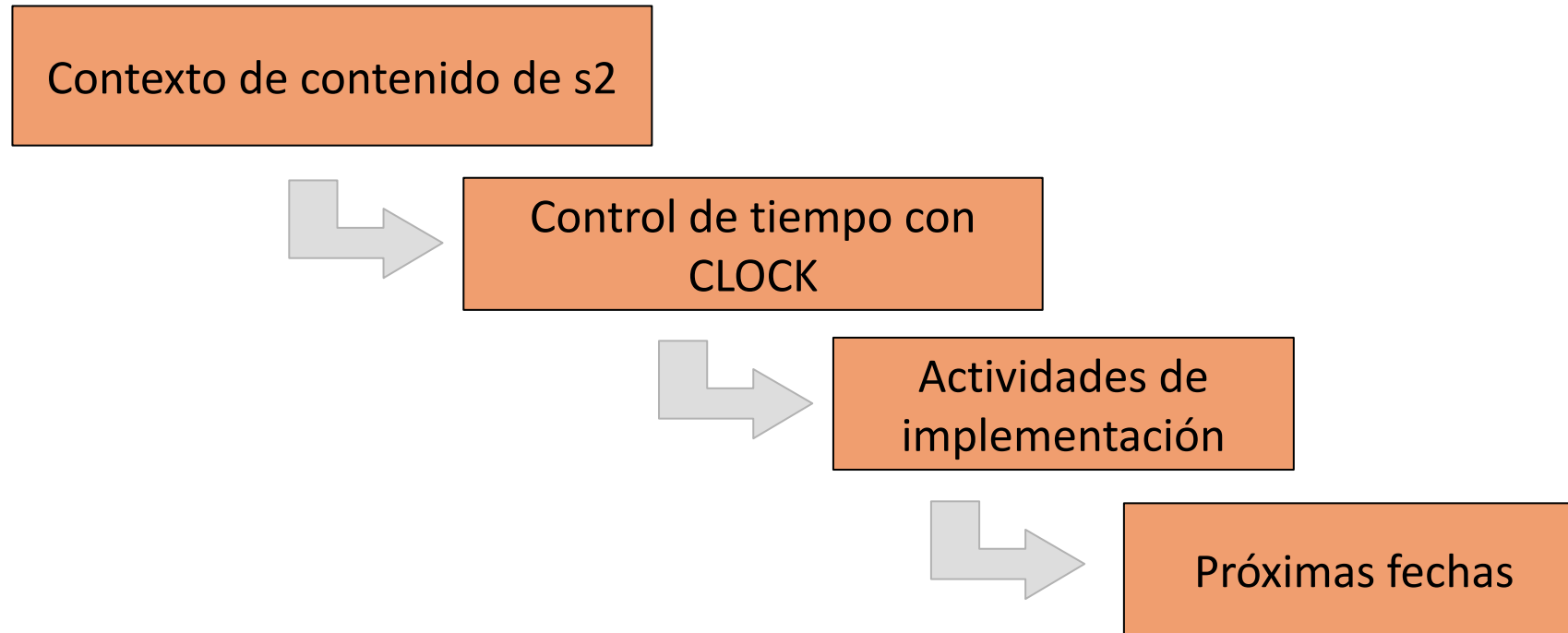
- Medición de tiempos de ejecución en C

- **Objetivos:**

- Evaluar empíricamente implementaciones en C de algoritmos iterativos
- Evaluar empíricamente implementaciones en C de algoritmos recursivos

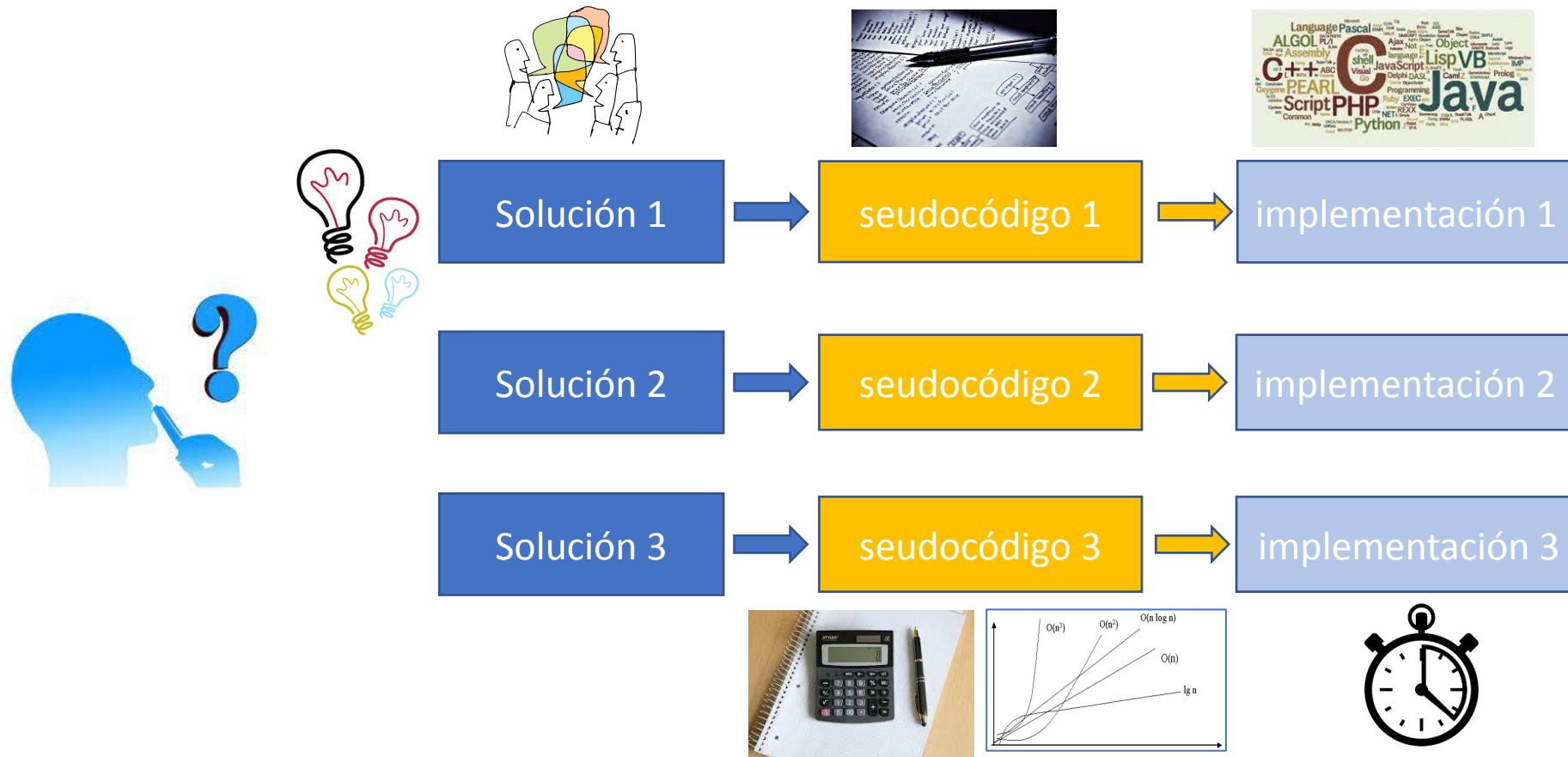


Ruta de trabajo





Contexto de semana 2



¿Los cálculos teóricos realizados para la complejidad algorítmica se ven reflejados de forma empírica?



Control de tiempo con *clock*

```
int main (int argc, char * argv[])
{
    int base=atoi(argv[1]);
    int exponente=atoi(argv[2]);
    long potIt=0, potRe=0;
    clock_t ciniIt, cfinIt, ciniRe, cfinRe;
    double tiempoIt=0.0, tiempoRe=0.0;

    ciniIt=clock();
    //potIt=potenciaIterativa(base,exponente);
    cfinIt=clock();

    ciniRe=clock();
    //potRe=potenciaRecursiva(base,exponente);
    cfinRe=clock();

    tiempoIt= (double)(cfinIt-ciniIt)/CLOCKS_PER_SEC;
    tiempoRe= (double)(cfinRe-ciniRe)/CLOCKS_PER_SEC;
}
```

- Biblioteca requerida: **time.h**
- **clock (void)** Devuelve el número de pulsos de reloj desde que se inició el proceso. Se almacena en una variable de tipo **clock_t**
- Para calcular el tiempo transcurrido se debe hacer la diferencia entre dos mediciones.
- **CLOCKS_PER_SEC**: Constante que define el número de pulsos de reloj por segundo.



Actividad de implementación 1

1. Implementar la siguiente función en C

```
long potenciaIterativa(int base, int exponente);
```

que devuelva el resultado de $\text{base}^{\text{exponente}}$ de forma iterativa

2. Evaluar la función creada, generando llamada desde función *main()* en lab02-tiempos.c



Actividad de implementación 2

1. Implementar la siguiente función en C

```
long potenciaRecursiva(int base, int exponente);
```

que devuelva el resultado de $\text{base}^{\text{exponente}}$ de forma recursiva

2. Evaluar la función creada, generando llamada desde función *main()* en *lab02-tiempos.c*



Actividad de implementación 3

1. Quitar comentarios extendidos en función *main()* de lab02-tiempos.c
2. Incorporar el control de tiempo (con clock) antes y después de llamar a las funciones *potencialterativa* y *potenciaRecursiva*
3. Desplegar resultados por consola para cálculo de potencia iterativo y recursivo, indicando la base, el exponente, el resultado y el tiempo de ejecución (para cada caso)



Actividad de implementación 4

1. Identificar valores de base y exponente, para los cuales hay diferencia en tiempo de ejecución de versión iterativa y recursiva de cálculo de potencia
2. Evaluar con valores de base y exponentes mayores identificados en paso anterior
3. Si se grafican tiempos obtenidos, se corresponden con valores de complejidad teórica?



Actividad de implementación 5

1. Escribir resultado de versión iterativa en el archivo de salida “potencialterativa.out”, indicando en una misma línea el valor de base, valor de exponente, valor de potencia y tiempo de ejecución.
2. Escribir resultado de versión recursiva en el archivo de salida “potenciaRecursiva.out”, indicando en una misma línea el valor de base, valor de exponente, valor de potencia y tiempo de ejecución



Entrega de actividad de laboratorio

- Entrega obligatoria
- Subir SOLO actividades 1, 2 y 3 de esta sesión en buzón de uVirtual, en único archivo **s2_apellido_nombre.zip**
- Plazo: **hoy** dentro del horario de laboratorio

Actividad de cierre



- Ir a menti.com e ingresar código 2718 1385



<div>  Octubre 2023 </div>						
Domingo	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
1	2	3 	4 	5	6 	7
8	9 	10	11	12 	13 	14
15	16 <small>Encuentro de Dos Mundos</small>	17	18	19 	20 	21
22	23	24	25 	26 	27  <small>Día Nacional de las Iglesias Evangélicas y Protestantes</small>	28
<div> <div></div> <div>Receso</div> </div>						