

Clase a Distancia

Estructuras de Selección, armado de Equipos, y Trabajo #1.

Clase a Distancia

- Leer sobre estructuras de control de la página 19 de <https://droscarbruno.files.wordpress.com/2015/02/modulo1introduccion20151.pdf>
- Armar equipos de trabajo de cinco integrantes como mínimo, enviar los nombres y usuarios GitHub a UTNFRBAAED@yahoogroups.com, antes del Lunes
- Resolver en equipo el trabajo #2: Mayor de dos números
Dado dos números informar cuál es el mayor.
Se debe entregar el análisis del problema, el algoritmo y la implementación.
 - <https://josemariasola.wordpress.com/aed/assignments/>
La fecha de entrega es antes de Semana Santa.
 - Leer los requisitos de entrega de <https://josemariasola.wordpress.com/aed/assignments/> como todavía son los primeros trabajos, hay algunos requisitos que no van a poder cumplir, intenten cumplir la mayoría.

Clase #03 de 27

Repaso, Selección, Notaciones Algorítmicas

Abril 20, Jueves

Agenda para esta clase

- Repaso de clases anteriores: [#1](#), [#2](#), y [clase online](#)
- Modelo IPO para Mayor
- Selección & Notaciones Algorítmicas
- Variantes de Mayor.cpp
- Abstracción del Control de Flujo de Ejecución mediante Estructuras

Modelo IPO para Mayor

Modelo IPO para Mayor

- Identificar entradas: fácil
 - Dos números
- Identificar salidas:
 - ¿Un número?
 - ¿Un texto?
 - Σ^*
 - ¿Un texto y un número?
 - Unión
 - ¿Y si ambos son iguales?

Selección & Notaciones Algorítmicas

Selección & Notaciones Algorítmicas

- Léxico
 - Define las entidades que manipula el algoritmo
 - Usa notación matemática, no usa un lenguaje de programación
 - El mismo para todas las notaciones
- Secuencial con saltos condicionales
 - Textual
 1. Leer a
 2. Leer b
 3. Substraer a de b
 4. Si positivo ir a 8
 5. Si negativo ir a 10
 6. Mostrar "Iguales"
 7. Fin
 8. Mostrar "b"
 9. Fin
 - 10. Mostrar "a"
 - Concepto de precondición
- Visual
 - Abstracto
 - Pasos y bifurcaciones
 - Diagrama de Flujo
- C++
 - Etiqueta
 - Sentencia goto
- Estructurada
 - Textual con indentación
 - Pseudocódigo con indentación
 - Visual
 - Chapin / Lindsay / UTN FRBA
 - Nassi-Shneiderman
 - C++
 - Sentencia if
- Ventajas y Desventajas.

Mayor de dos números

```
/* JMS
 * 20160411
 * Mayor de dos números
 * Muestra frase.
 */
#include <iostream>

int main() {
    int a, b;

    std::cin >> a;
    std::cin >> b;

    if( a > b )
        std::cout << "El mayor es a.\n";
    else if( b > a )
        std::cout << "El mayor es b.\n";
    else
        std::cout << "Son iguales.\n";
}
```

Mayor de dos números, v2

```
/* JMS
 * 20160411
 * Mayor de dos números
 * Muestra frase y valor.
 */
#include <iostream>

int main() {
    int a, b;

    std::cin >> a;
    std::cin >> b;

    if( a > b )
        std::cout << "El mayor es: " << a << "\n";
    else if( b > a )
        std::cout << "El mayor es: " << b << "\n";
    else
        std::cout << a << " y " << b << " son iguales.\n";
}
```

Mayor de dos números, v3

```
/* JMS
 * 20160411
 * Mayor de dos números
 * Muestra valor o frase.
 */
#include <iostream>

int main() {
    int a, b;

    std::cin >> a;
    std::cin >> b;

    if( a > b )
        std::cout << a << "\n";
    else if( b > a )
        std::cout << b << "\n";
    else
        std::cout << "Son iguales.\n";
}
```

Mayor de dos números, v3 revisado

```
/*
 * JMS
 * 20160411
 * Mayor de dos números.
 * Operador igualdad y refactor de salidas. ¿Funciona?
 */
#include <iostream>

int main() {
    int a, b;

    std::cin >> a;
    std::cin >> b;

    if( a == b )
        std::cout << a << " y " << b << " son iguales.";
    else
        std::cout << "El mayor es: ";
        if( a > b )
            std::cout << a;
        else
            std::cout << b;
    std::cout << "\n";
}
```

Mayor de dos números, v5

```
/*
 * JMS
 * 20160411
 * Mayor de dos números.
 * Operador igualdad y refactor de salidas.
¿Funciona?
*/
#include <iostream>
int main() {
    int a, b;
    std::cin >> a;
    std::cin >> b;
    if( a == b )
        std::cout << a << " y " << b << " son iguales.";
    else{
        std::cout << "El mayor es: ";
        if( a > b )
            std::cout << a;
        else
            std::cout << b;
    }
    std::cout << "\n";
}
```

- Agrupamiento
- Llave
- Bloque
- Sentencia compuesta

Abstracción del Control de Flujo de Ejecución mediante Estructuras

¿Cómo ayudamos a Bart a escribir la frase 42 veces?

Voy a practicar programación *Voy a practicar programación*
Voy a practicar programación
Voy a practicar programación
Voy a practicar programación
Voy a practicar programación



Abstracción del Control de Flujo de Ejecución mediante Estructuras

- Secuencia
 - Orden de los pasos
- Selección
 - Condición
 - Decisión
 - Bifurcación
- Iteración
 - Condición
 - Repetición
 - Bifurcación

Términos de la clase #03

Definir cada término con la bibliografía

- Modelo IPO para Mayor
 - Conjunto de valores a los que pertenecen las entradas y salidas
 - Caracteres: Σ
 - Cadenas: Σ^*
- Selección & Notaciones Algorítmicas
 - Léxico
 - Notación textual con saltos condicionales
 - Precondición
 - Etiqueta
 - Sentencia goto
 - Representación estructurada
- Pseudocódigo
- Chapin / Lindsay / UTN FRBA
- Nassi-Shneiderman
- Sentencia if-else
- Variantes de Mayor.cpp
 - Operador igualdad
 - Bloque
 - Sentencia compuesta
- Abstracción del Control de Flujo de Ejecución mediante Estructuras
 - Secuencia
 - Selección
 - Iteración.

Tareas para la próxima clase

1. Trabajo #3 — Repetición de Frase

- Frases
 - Saltos condicionales
 - Iteración Estructurada



¿Consultas?



Fin de la clase

Clase #04 de 27

Iteraciones & Tipos de Datos

Abril 27, Jueves

Agenda para esta clase

- Iteraciones
- Tipos de Datos



Iteraciones

<https://josemariasola.wordpress.com/aed/papers/#Iterations>

Introducción a Tipos de Datos

En Matemática y en C++

¿Qué es un Tipo de Dato?

- Buscar una definición
- Buscar ejemplos
- Representación
- Valores que almacena
- Operaciones que se pueden realizar
- Definición
 - Un *tipo de dato*, o simplemente tipo, es un *conjunto de valores* y un *conjunto de operaciones* sobre esos valores.

Tipos de Datos – Conjunto de Valores

Niveles de abstracción y Ejemplos

Matemática		C++		Microporcesador (general)
Conjunto	Valores	Tipo de Dato	Valores	Representación electrónica
\mathbb{B}	V F	bool	true false	Registro: Bit
Σ	A B N ! ... (finito) \approx ASCII	char	'a' 'A' '1' '+' '@' '\'' ...	Registro: Byte
\mathbb{N}	Semirecta	unsigned	0 1 2 96 ...	Registro: Word
\mathbb{Z}	Recta	int	1 -2 1024 ...	Registro: Word
\mathbb{R}	Recta	double	1.5 1.0 1e5 ...	Registro: Floating point 80-bits
Σ^*	A ABA AÑABA ...	string	"a" "aA" "abab" "" ...	Bloque contiguo de memoria

Tipos de Datos – Conjunto de Operaciones

Niveles de abstracción y Ejemplos

Matemática		C++	
Tipo	(Algunas) Operaciones	Tipo	(Algunas) Operaciones
\mathbb{B}	$= \neq \vee \wedge \neg$	bool	$== != \text{ or } \text{ and } \text{ not}$
Σ	$= \neq$	char	$== != <= >= + - * / - \%$
\mathbb{N}, \mathbb{Z}	$= \neq \leq \geq + - \cdot / - \text{ resto}$	unsigned int	
R	$= \neq \leq \geq + - \cdot / -$	double	$== != <= >= + - * / -$
Σ^*	$= \neq \cdot $	string	$== != <= >= + \text{ length}$

Tipos de Datos más Abstractos

- Ejemplos Comunes
 - String
 - Date
 - Money o Currency
 - Números Complejos
 - Compuestos
- ¿Qué característica comparten?
- No tienen representación directa en máquina
 - Un int puede ser almacenado en un registros del microprocesador, un string no.
- No son estándar en los lenguajes de programación

Tareas para la próxima clase

1. Trabajo #4 – Ejemplos de Valores y Operaciones de Tipos de Datos

Términos de la clase #04

Definir cada término con la bibliografía

- Iteraciones
 - Repetición
 - Diagrama de Flujo
 - Diagrama Nassi-Shneiderman (NS)
 - Pseudocódigo
 - Assembler
 - Generación de código desde diagrama NS
 - Eficacia
 - Eficiencia
 - Atributos de calidad
 - Etiqueta
 - Sentencia Goto
 - una-ó-más-repeticiones.
- Tipos de Datos
 - Definición
 - \mathbb{B}
 - Σ
 - N
 - Z
 - R
- Σ^*
- bool
- char
- unsigned
- int
- double
- string
- Representación literal o constante de valores
- Operaciones
- Resto
- Módulo
- Tipo de Datos más Abstractos



¿Consultas?



Fin de la clase

Clase #05 de 27

Eficiencia, Variables, & Tipos

Compuestos

Mayo 4, Jueves

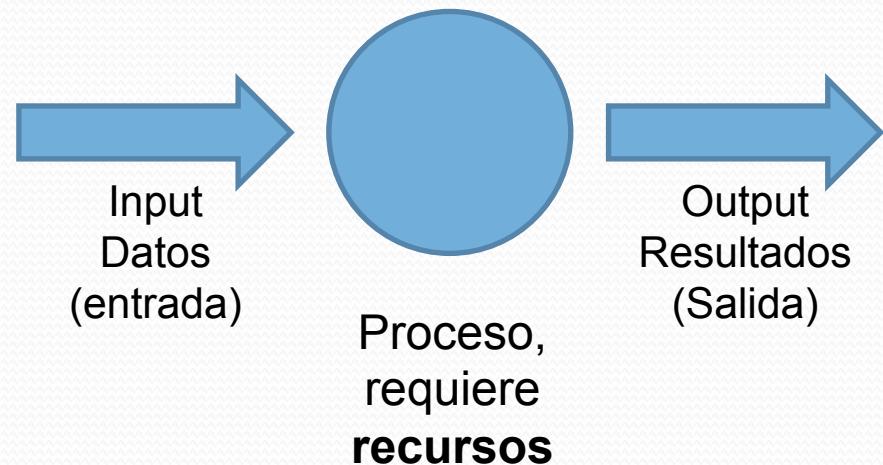
Agenda para esta clase

- Repaso de clase #4
 - Iteraciones
 - Tipos de datos
- Modelo Computacional
- Eficiencia
- Variables, Declaraciones, y Asignaciones
- Tipos de datos Compuestos

Modelo Computacional

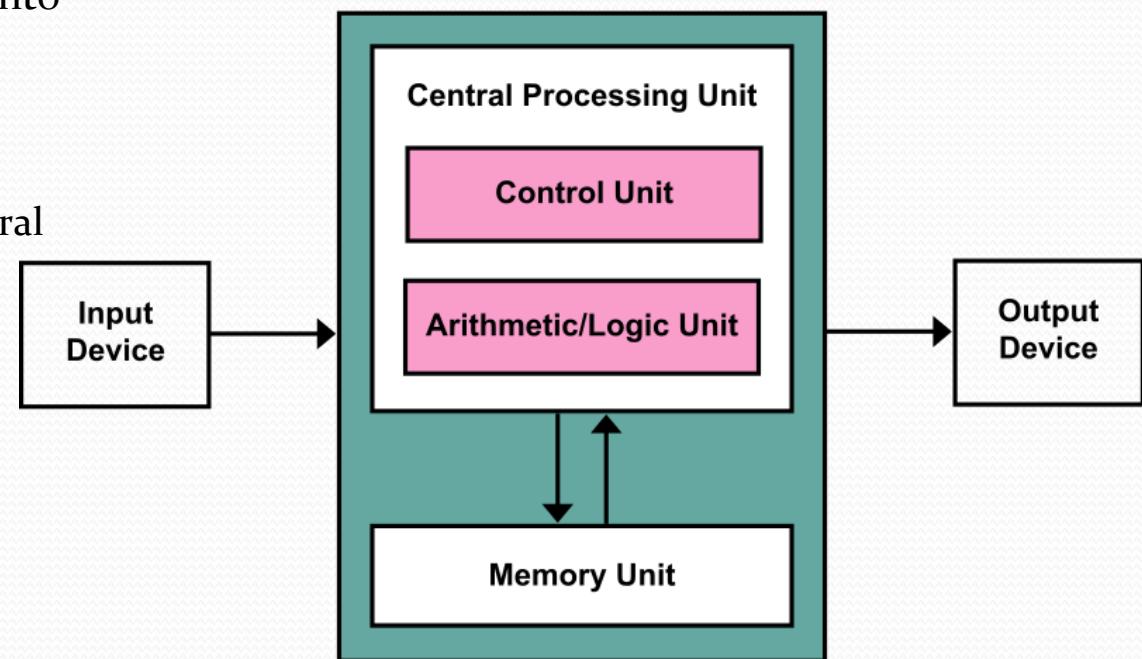
Modelo Computacional

- Modelo de procesamiento
 - Entrada
 - Proceso
 - Salida (y Errores)
 - Modelo IPO
- Arquitectura de von Neumann
 - John von Neuman
 - Computadora con unidad central de procesamiento que almacena programas y datos en memoria
- Ejemplos
 - Hola.cpp
 - Adición.cpp



Componentes de la Arquitectura von Neumann

- Unidad Central de Procesamiento
 - Registros
 - IR: Registro de instrucción
 - PC: Contar de Programa
 - CA: Aritmético Lógica
 - CC: Unidad de Control Central
- M: Memoria (Principal)
 - Datos
 - Programas
 - Lenguaje Máquina
- I: Entrada
- O: Salida
 - Teclado
 - Pantalla
- R: Memoria Secundaria
- Pros y Cons de Memoria Principal y Secundaria
 - Costo
 - Eficiencia
 - Tamaño
 - Persistencia



Comparación de Recursos Utilizados durante el Procesamiento

Espacio & Tiempo

Comparación Subjetiva

- Legibilidad
- Claridad
- Mantenibilidad
- Los adjetivos "idad" son los atributos de calidad del desarrollo.

```
//1           cout << c;  
int a;       //4  
int b;       int a;  
cin >> a;    int b;  
cin >> b;    cin >> a;  
cout << a+b; cin >> b;  
  
//2           int c = a+b;  
int a, b;    cout << c;  
cin >> a >> b; //5  
cout << a+b;  
  
//3           int a;  
int a;       int b;  
int b;       cin >> a;  
int c;       cin >> b;  
cin >> a;    b = a+b;  
cin >> b;    cout << b;  
c = a+b;
```

Recursos Utilizados – Comparación Objetiva

Durante	Recurso	
	Espacio	Tiempo
La traducción del Programa	de memoria secundaria que ocupa el programa fuente; en general, medido en líneas de código (lines of code, LOC).	requerido por el compilador para la traducción del programa en lenguaje de alto nivel a lenguaje máquina.
La Ejecución del Proceso	de memoria principal utilizado durante la ejecución del proceso de memoria secundaria utilizada como área de trabajo.	requerido para finalizar la ejecución del proceso.

Comparación Objetiva

- Tiempo
 - Las declaraciones no insumen tiempo de ejecución, salvo cuando tienen inicialización de una variable automática
 - Cantidad de operaciones (aproximación)
 - >>
 - <<
 - +
 - =
- Espacio
 - Cantidad de variables

```
//1
int a;
int b;                                //4
cin >> a;
cin >> b;
cout << a+b;

//2
int a, b;
cin >> a >> b;
cout << a+b;

//3
int a;
int b;
int c;
cin >> a;
cin >> b;
c = a+b;
cout << c;

//5
int a;
int b;
cin >> a;
cin >> b;
b = a+b;
cout << b;
```



Intervalo

20 minutos

Variables, Declaraciones, & Asignación

Variable, Valor, y Declaración

- Valor como dato
- Variable
 - “Valor nombrado”
 - Relación nombre-valor
 - Relación nombre-valor dinámico
 - Relación nombre-espacio de memoria
 - Relación nombre-espacio de memoria que puede almacenar diferentes valores en función del tiempo
 - Durante la ejecución del programa
 - Dinámico
 - $\text{valor}(\text{nombre}, \text{tiempo})=v$
 - Constante
 - $\text{dirección}(\text{nombre})=k$
- Declaración
 - Es un "Anuncio"
 - Sintaxis y Semántica
 - Matemática
 - $v \in T$
 - C++
 - $T \ v;$
 - Ejemplo
 - Matemática
 - $i \in \mathbb{Z}$
 - C++
 - $\text{int } i;$

Asignación

- $x \leftarrow 1$
 - Semántica
 - Secuencia
 - Pre condición
 - Pos condición
 - $Variable \leftarrow Valor$
 - Valor literal
 - Otra variable
 - Resultado de Expresión
 - Sintaxis
 - $x \leftarrow 1$
 - $x := 1$
 - $x = 1$
 - Tipos de asignación
 - Explícita o Interna
 - Desde flujo de entrada o Externa
 - Asignación vs. Igualdad
 - Declaración con inicialización.
- $x = 1;$
 - $cin >> x;$
 - $x == 2$
 - $int x = 42;$

Tipo de Dato Compuesto

Tipo: Aula

- Conjunto de Valores: Representación
 - Natural
 - Restricciones asociadas
 - Problema 1
 - Encontrar el aula
 - Contexto
 - Problema 2
 - Encontrar un aula con suficiente capacidad, proyectos, ventilación
 - Conecto
 - Solución
 - Tupla

Términos de la clase #05

Definir cada término con la bibliografía

- **Modelo Computacional**
 - Arquitectura de von Neumann
 - Recursos
 - Memoria para datos y programas
 - Registros
 - Memoria principal
 - Memoria secundaria
- **Eficiencia**
 - Tiempo
 - Espacio
- Durante traducción
- Durante ejecución
- Atributos de calidad
- **Variables, Declaraciones, y Asignaciones**
 - Definición de Valor
 - Definición de Variable
 - Definición de Declaración
- Asignación
- Precondición
- Poscondición
- Diferencias con igualdad
- Declaración con inicialización.
- **Tipos de datos Compuestos**
 - Tupla ó n-upla
 - Restricciones entre componentes

Tareas para la próxima clase

1. Trabajo #5 — Especificación del Tipo de Dato Fecha
 - Conjunto de Valores
 - Conjunto de Operaciones.



¿Consultas?



Fin de la clase