

Algorítmica y Programación II

Tipo de dato abstracto

TDA

- › Es un tipo de dato **definido por el programador**
 - Especificación de la **representación de los elementos** del tipo de dato. (organización y rango de valores de los elementos).
 - Especificación de las **operaciones** permitidas con ese tipo.
 - **Encapsulamiento** de lo anterior, de manera que el usuario no pueda manipular los datos del objeto, excepto para el uso de las operaciones

TDA

Conceptualmente hasta ahora teníamos:

programa = datos + **algoritmos**

A partir de la incorporación del TDA:

algoritmos = alg. de datos + alg. de control

programa = **datos** + alg. de datos + alg. de control

Dado la definición de TDA se establece:

Programa = { TDA + algoritmo de control }

TDA

› Ejemplos

- Los Conjuntos, con sus operaciones.
- Los números complejos.
- Las matrices cuadradas (operaciones como Suma, Resta, Multiplicación y Traspuesta).
- Las pilas, las colas...etc..

TDA

› Características:

- Lleva en si mismo la representación y el comportamiento de sus **objetos**.
- Permite **desarrollar y probar su código de manera independiente** del programa que lo va a utilizar.
- Es **independiente** del programador y del programa que lo utilice.
- La independencia anterior garantiza su **reusabilidad**.

TDA

› Ventajas:

- El programa o módulo que referencia a un TDA , lo **utiliza como si fuera una “caja negra”**, de la que se obtienen resultados a través de operaciones definidas
- Permite que nos concentremos en la implementación del TDA y no en el programa.
- No interesa saber cómo funciona internamente, sólo se necesita conocer las operaciones que nos ofrece.

TDA

› Ventajas:

- El programa o módulo que referencia a un TDA , lo **utiliza como si fuera una “caja negra”**, de la que se obtienen resultados a través de operaciones definidas
- Permite que nos concentremos en la implementación del TDA y no en el programa.
- No interesa saber cómo funciona internamente, sólo se necesita conocer las operaciones que nos ofrece.

TDA

› Conclusiones

- A la idea de **Abstracción algorítmica** (dividir el problema en subproblemas de menor complejidad) se **agrega** la noción de **abstracción de tipos de datos**, buscando generalizar las soluciones para lograr **reusabilidad**.
- **Código más fácil de verificar y mantener**, aunque su creación implique un mayor tiempo de desarrollo y una posible pérdida de eficiencia en una aplicación particular.

TDA

› Implementación en Pascal:

- El lenguaje Pascal no permite una total implementación de un TDA. (Como veremos está fuertemente ligado a la especificación del tipo de dato).
- Haremos una implementación Intermedia utilizando las **unidades externas definidas por el programador (UNIT)**.
- El lenguaje ADA permite una implementación total del TDA.

TDA – implementación

unit nombre;

interface

} pública

implementation

} privada

begin

 //inicialización

end.

TDA – ejemplo racionales

unit **uracional**;

interface

type

 TRacional = record

 num,

 den: integer;

 end;

procedure leer(var R:TRacional);

procedure imprimir(const a:TRacional);

procedure sumar(const a,b:TRacional; var c:TRacional);

TDA – ejemplo racionales

...

implementation

```
procedure leer(var R:TRacional);  
begin  
    write('ingrese numerador y denominador');  
    read(r.num, r.den);  
end;
```

```
procedure imprimir(const a:TRacional);  
begin  
write(a.num, '/', a.den);  
end;
```

TDA – ejemplo racionales

...

implementation

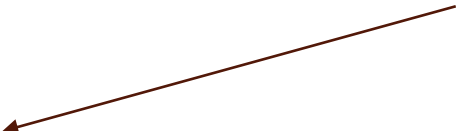
```
procedure sumar(const a,b:TRacional;  
                var c:TRacional);  
  
begin  
  with c do  
    begin  
      den := a.den * b.den;  
      num := (den div a.den * a.num) +  
             (den div b.den * b.num);  
    end;  
end;
```

TDA – ejemplo racionales

```
program Racionales;
uses crt, uRacional;
var a,b : TRacional;
    res : TRacional;

begin
    leer(a);
    leer(b);
    sumar(a,b,res);
    imprimir(res);
end.
```

uRacionales.leer(a);



TDA - Bibliografía

- Introducción a la programación con pascal. Rafael, Llavori.
- Algoritmos, Datos y programas. Armando, De Gusti.
- Estructuras de datos y algoritmos. Mark, Weiss.
- https://es.wikipedia.org/wiki/Tipo_de_dato_abstracto