



**Universidad de Matanzas**  
**Camilo Cienfuegos**  
Facultad de Informática

**Título:**

**La tarea extraclase:** Una posibilidad para la creación de productos informáticos y una opción para la interdisciplinaridad.

**Autores:**

Antonio C. Fernández Orquín  
Roger Pérez Chávez  
Katia Vila Rodríguez  
Iván Pérez Mallea. (ISPJAE).  
Arsenio Miguel Iznaga Benítez. (ISPJAE).  
Walfredo González

## **Tabla de contenidos**

Resumen: .....	3
Introducción .....	4
Desarrollo .....	6
Conclusiones .....	14
Bibliografía .....	15

## **Resumen:**

Se han recogido las experiencias desarrolladas en la vinculación de varias asignaturas del perfil del Ingeniero Informático, a través de una tarea integradora, la cual permite al estudiante reutilizar y elaborar nuevos elementos en materia de conceptos y conocimiento. Se ofrece una panorámica del proceso de integración de las disciplinas involucradas en el estudio realizado. Se describe cómo, a través de la tarea extractases, se puede incentivar la mentalidad desarrolladora del Ingeniero Informático, así como los procesos lógicos en la creación de aplicaciones informáticas. Se ofrecen algunos ejemplos de las aplicaciones elaboradas por los estudiantes de segundo año de la carrera. Se plantean las ideas seguidas para resolver la carencia de software para la enseñanza de algunas asignaturas, que no son propiamente de la facultad de Informática.

## Introducción

Una necesidad histórica de la enseñanza ha sido siempre la lucha por la interrelación entre las diferentes asignaturas del perfil profesional de los alumnos que estudian en nuestras universidades. Hoy en día cada vez se hace más importante poder estrechar los vínculos entre las diferentes asignaturas que se imparten. Las disciplinas básicas no pueden estar divorciadas, en especial la computación. Llevarlas al estudiantado totalmente aisladas sería como darles todas las piezas y no la metodología para ensamblarlas. Un enfoque de un contenido específico de una determinada asignatura, desde la óptica de otra que lo utiliza, ayuda a comprender la aplicación al mundo real de muchos conocimientos, que vistos sólo desde la asignatura que los contiene e imparte, en ocasiones se quedan en un nivel muy abstracto. (Fernández, 2004).

Surgido de la unión de las tecnologías de la computación y la televisión, el campo de la gráfica por computadora ha sido explorado desde hace muy poco tiempo y si bien hemos visto un gran desarrollo en este sentido en los últimos años, no debemos olvidar que la verdadera dificultad en la creación de gráficos estriba en la compleja naturaleza del tema. El estudio de la gráfica por computadora exige una comprensión de la máquina y las tecnologías de exhibición y una base sólida en geometría analítica, trigonometría y otras ramas de la matemática, así como un dominio amplio de las técnicas de programación. Además son necesarios los conocimientos sobre operaciones con matrices, estructuras de datos, tratamiento de ficheros y algoritmización. Si tomamos en consideración estas palabras, entonces tendremos que ser capaces de entrelazar armónicamente los conocimientos de todas estas asignaturas y así hacerlos llegar al estudiantado.

La tarea es un recurso importante para la asignatura Gráfica por Computadora (GPC) y Diseño Asistido por Computadoras (DAC), ya que estas no tienen examen final. Hemos tratado de utilizar este recurso docente para que los estudiantes demuestren, además de los conocimientos de la teoría de los

gráficos por computadoras, también los de la disciplina programación, matemática y la asignatura geometría descriptiva (esta no es impartida en el currículo del Informático). A través del desarrollo de la tarea hemos tratado de implementar la secuencia de pasos del ciclo del desarrollo de software y con esto hacer un vínculo con la disciplina Ingeniería de Software. Como un resultado importante de todo esto se han elaborado varios software que servirán como herramientas de apoyo a la enseñanza de la geometría descriptiva para los Ingenieros Mecánicos.

Surgido de la unión de las tecnologías de la computación y la televisión, el campo de la gráfica por computadora ha sido explorado desde hace muy poco tiempo y si bien hemos visto un gran desarrollo en este sentido en los últimos años, no debemos olvidar que la verdadera dificultad en la creación de gráficos estriba en la compleja naturaleza del tema. El estudio de la gráfica por computadora exige una comprensión de la máquina y las tecnologías de exhibición y una base sólida en geometría analítica, trigonometría y otras ramas de la matemática, así como un dominio amplio de las técnicas de programación. Además son necesarios los conocimientos sobre operaciones con matrices, estructuras de datos, tratamiento de ficheros y algoritmización. Si se toman en consideración estas palabras, entonces es preciso tener la capacidad de entrelazar armónicamente los conocimientos de todas estas asignaturas y así hacerlos llegar al estudiantado.

Este trabajo se propone como objetivo fundamental contar un conjunto de experiencias de los autores, en la utilización de la tarea de las asignaturas Gráfica por Computadoras (GPC) y Diseño asistido por Computadoras (DAC), para el desarrollo de software educativo y la interrelación de los contenidos de varias asignaturas del perfil de Ingeniero Informático.

## Desarrollo

Instrumentar un recurso educativo, como lo es la tarea, no es una labor sencilla; máxime si queremos lograr resultados tan exigentes como los propuestos en este trabajo. Para comenzar esta instrumentación lo primero que se analizó fue la posibilidad de generar variantes que respondieran a una problemática real y no ha simples ejemplos docente.

Luego de un estudio de la situación de la computación en la carrera de Mecánica, corroborado a través de los exámenes de control de las habilidades de Computación, que propone el Ministerio de Educación, se detectaron algunos problemas:

Los estudiantes de tercero y cuarto año presentan problemas con el uso de bases de dato y con la utilización de entrenadores y tutoriales informáticos que los ayuden en la autopreparación de las asignaturas que reciben. Analizado esto nos dimos a la tarea de incorporar en las variantes de la tarea de GPC y DAC, ejercicios que pudieran ser utilizados por profesores y alumnos vinculados a la geometría descriptiva, asignatura que se imparte en los primeros años de varias ingenierías.

Un elemento importante para la solución de problemas de la Computación Gráfica es la programación. En este caso se ha hecho un trabajo encaminado a vincular la disciplina Programación, con GPC y DAC. Un ejemplo de esta necesidad se puede apreciar en el tema I. Proyección y trazado de gráfica se necesita que el alumno elabore algoritmos que sean capaces de trazar en el plano (2D): puntos, segmentos de recta, circunferencias, elipses, rectángulos, etc. (Iznaga, 2002). Para cumplir con el postulado anterior, de la asignatura Programación I, se aprovechan las clases creadas TPunto, TFigura, TSegRecta, TPoligonos, etc. En ellas están encapsuladas las herramientas que nos permitirán cumplir con una parte importante de los objetivos trazados. Esto surge de un estudio de los autores y una coordinación en la impartición de las asignaturas de forma que se garantice este objetivo. También se vieron otros elementos pertenecientes a las asignaturas Programación II.

## ***La Gráfica por Computadora y la Programación.***

Hoy en día cada vez se hace más importante poder estrechar los vínculos entre las diferentes asignaturas que se imparten en nuestras universidades, las disciplinas básicas no pueden estar divorciadas, en especial la computación, llevarlas al estudiantado totalmente aisladas sería como darles todas las piezas y no la metodología para ensamblarlas. (Fernández, 2002)

A nuestro juicio ambas asignaturas tienen muchos puntos de contacto que mostraremos a continuación:

La asignatura Programación I es impartida en primer año, los estudiantes reciben el paradigma orientado a objeto. La asignatura Gráfica por Computadora se imparte en el 2do semestre de segundo año, lo que nos permite recibir un estudiante con preparación para enfrentar las tareas de programación que demanda esta asignatura.

Por otra parte la forma en que se ha impartido la Programación I nos permite aprovechar muchos recursos para cumplir los objetivos de Gráfica por computadora. Pongamos un ejemplo.

En el tema I. Proyección y trazado de gráfica por computadora se necesita que el alumno elabore algoritmos para trazar en el plano (2D): puntos, segmentos de recta, circunferencias, elipses, rectángulos, etc. (Iznaga, 2002)

Para cumplir con el postulado anterior, de la asignatura Programación I se aprovechan las clases ya creadas: TPunto, TFigura, TSegRecta, TPoligonos, etc. En ellas están encapsuladas las herramientas que nos permitirán cumplir con una parte importante de nuestros objetivos.

El tema I también comprende la determinación de la posición relativa de la recta respecto al plano, distancia entre punto y recta, entre rectas, situación del punto con respecto a la recta, etc. Todos estos elementos son colocados como métodos de las clases anteriores, haciendo aún más robustas dichas estructuras, facilitando con ello la creación de programas utilizando estas clases. Ejemplo de lo antes mencionado lo

constituye el método "Distance" de la clase TPunto, el cual nos permite determinar la distancia entre dos puntos.

```
Function TPunto.Distance(P2:TPunto):Single;  
Begin  
    Result:= Sqrt(Sqr(P2.X-X) + Sqr(P2.Y-Y));  
End;
```

Nuestros estudiantes harán uso de todas las posibilidades de la POO, tienen que usar clases ya elaboradas con anterioridad por otras asignaturas precedentes, declarar nuevas clases, crear objetos utilizar la herencia y el polimorfismo. Analizando lo antes mencionado podemos ver que existe una estrecha vinculación entre las asignaturas Programación I y Gráfica por computadora.

La asignatura programación II es impartida en el primer semestre de 2<sup>do</sup> año, cuestión esta que también nos favorece grandemente, pues en ella el alumno aprende a trabajar con estructuras de datos más complejas como son los arreglos dinámicos y las listas, estructuras ampliamente utilizadas en Gráfica.

Un ejemplo de puntos de contactos entre ambas asignaturas lo es la representación de Polilíneas y Polígonos, en los que se hace necesario almacenar una colección de puntos.

Veamos un ejemplo del método dibujar de la clase TPolilinea creado por un estudiante nuestro:

```
Procedure TPolilinea.Dibujar;  
    Var i: integer;  
    Begin  
        i:=0;  
        Area.Canvas.Pen.Color := Color;  
        Area.Canvas.Pen.Style:= MyStyle;  
        Area.Canvas.MoveTo(Puntos[i].X, Puntos[i].Y);  
        While (i < length(puntos)-1) do  
            Begin  
                Area.Canvas.LineTo(Puntos[i + 1].X, Puntos[i + 1].Y);  
                inc(i);  
            End;  
    End;
```

Como se puede apreciar Puntos es un Arreglo dinámico y si se observa su declaración se tendrá algo así:

```
TPuntos = Array of TPoint;
```

TPoint esta definida por Borland Delphi como:

**Type**

```
TPoint = packedrecord X: Longint; Y: Longint;End;
```

En otras implementaciones se verán estructuras del tipo Listas como es el caso:

```
TListaFiguras = Class (TList)  
  Procedure Dibujar;  
  Procedure Seleccionar (Micursor:TMycursor;Var Indice:Integer);  
End;
```

TListaFiguras es una estructura de datos de tipo colección o Lista de elementos que nos sirve para almacenar varias figuras, las que posteriormente serán utilizadas.

Se utilizan otras estructuras como la TObjectList que no es más que una lista para almacenar objetos. Ejemplo:

```
TPlano = class (TObjectList)  
  Area:TForm;  
  POne,PTwo:Integer;  
  fPoint:TPoint;  
  findex:Integer;  
  Seleccionado:Integer;  
  Directorio:String;  
  Constructor Create(f:TForm);  
  Procedure WriteAll;  
  Procedure Nuevo;  
  Procedure Eliminar (P:Integer);  
End;
```

Se han utilizado los llamados Procedimientos de objetos para la implementación de los eventos (De esta manera se crea un evento que se visualiza en el *Object Inspector* de *Borland Delphi*)

```
TonGraph = procedure (Sender: TGraphtool; Expresion: Shortstring) of object;
```

```
TonCenterChanged = procedure (Sender: TGraphtool; dX,dY: Integer) of object;
```

También se utiliza el tipo de dato puntero para la creación de estructuras tipo pila, en las que se guardan los gráficos.

```
TExpresiones = ^nodo;  
Nodo = record  
    Color :TColor;  
    Apunt :TExpresiones;  
    Info :ShortString;  
End;
```

Como se ha podido ver existen un gran número de situaciones en las que los estudiantes tienen que utilizar los contenidos de programación, se trabaja con estructuras de datos tales como: Clases, Registros, Arreglos, Listas y otras.

### ***¿Porqué utilizar la Programación Orientada a Objetos?***

Podríamos filosofar un rato y cubrir todo este artículo justificando la necesidad de utilizar la POO, pero nos parece más productivo ilustrarlo mediante dos elementos:

1. Los estudiantes de 2<sup>do</sup> año de informática en el momento que reciben la Computación Gráfica, ya han pasado por la programación I y II. Ambas utilizan el paradigma orientado a objeto. De manera que por inercia lógica debemos continuar con este estilo.
2. Se ilustrará con un ejemplo para que se vea mejor lo que se está defendiendo. Se hará utilizando *Object Pascal*, pero podría ser con cualquier lenguaje de programación:

Utilizando sólo los recursos que da *Borland Delphi* para trazar una línea con un color

```
Begin  
Canvas.pen.style:= psSolid;  
Canvas.Pen.Color:= clBleu;  
Canvas.MoveTo(40,40);  
Canvas.LineTo(100,100);  
End;
```

Utilizando un clase que fue creada en la asignatura programación I

```
Begin  
SegRecta:=TSegRecta.Create(40,40,100,100,clBlue,  
form1,psSolid);  
SegRecta.Dibujar;  
End;
```

¿Y qué pasaría si se desea saber la distancia que hay de un punto P3 a la recta que acabamos de trazar?

En el caso de la programación estructurada, se deja al lector para que lo resuelva y analice. Con la clase TSegRecta con la que se está trabajando sería:

```
Label1.caption:= floatToStr (SegRecta.Distancia (50,35));
```

Simplemente haciendo una llamada al método Distancia, implementado con anterioridad, se resolvería el problema y se podría asignar el valor obtenido a una etiqueta que muestre el texto como se muestra en la figura.

Para la implementación de la gran mayoría de los trabajos también es necesario tener dominio de varios elementos de Matemática, específicamente de Geometría Analítica y Matemática Numérica. Para esto se retoman los contenidos de Matemática II y IV. Se ha logrado una estrecha vinculación entres los profesores que imparten la Matemática y los de GPC y DAC.

### ***La vinculación de la tarea al proyecto del año***

Otro elemento importante para el desarrollo de esta experiencia fue la incorporación de la tarea al proyecto de Ingeniería de Software. Es evidente que para desarrollar un producto de calidad los estudiantes deben contar, no sólo con sus conocimientos, sino también necesitan de tiempo para realizar su trabajo. Con este objetivo se logró vincular el desarrollo de las aplicaciones realizadas con la fase de análisis y diseño del proyecto de Ingeniería. Esto provocó que se concentraran en un solo trabajo con el que obtendrían la nota de estas asignaturas. Obviamente los estudiantes desarrollarían un software para cumplimentar la tarea de DAC en un caso y otros la de GPC, a la que se le haría la Ingeniería para cumplimentar el proyecto de Ingeniería de Software. Posteriormente se discutirían juntos formando un tribunal en el que participaría los profesores de Programación, Gráfica e Ingeniería de Software.

Dentro de las asignaturas del perfil de muchas de la Ingenierías se encuentran la Geometría Descriptiva y el Dibujo, pero son precisamente éstas las que han tenido menos aplicación de la Computación. Como se ha visto con anterioridad se detectaron algunos problemas de utilización de software educativo en las asignaturas antes mencionadas. Por esta razón se han diseñado la mayoría de las variantes de las tareas de GPC y DAC en

función de obtener aplicaciones útiles para el aprendizaje de algunos temas de Geometría Descriptiva y Dibujo Básico, éstas se enumeran a continuación:

#### Para DAC

1. Representación del punto en un sistema ortogonal en el primer cuadrante. Con su representación isométrica.
2. Intersección entre un plano dado por sus trazas y un prisma recto de base triangular. Con la representación del truncamiento en isométrico.
3. Intersección entre un plano dado por sus trazas y una pirámides recta de base triangular. Con la representación del truncamiento en isométrico.
4. Intersección entre un plano dado por sus trazas y un prisma oblicuo de base triangular. Con la representación del truncamiento en isométrico.

#### Para GPC

1. Representación del punto en un sistema ortogonal en los 8 octantes.
2. Intersección entre un plano dado por sus trazas y un prisma recto de base triangular.
3. Intersección entre un plano dado por sus trazas y una pirámides recta de base triangular.
4. Intersección entre un plano dado por sus trazas y una pirámides oblicua de base triangular.
5. Intersección entre un plano dado por sus trazas y un prisma oblicuo de base triangular.
6. Intersección entre planos dados por sus trazas.
7. Intersección entre plano dado por un triángulo y uno dado por sus trazas.

8. Intersección entre rectas y planos dados por sus trazas.
9. Intersección entre rectas.
10. Determinación de la verdadera magnitud de la recta por método de giro.
11. Determinación de la verdadera magnitud de la recta por método de cambio de planos.
12. Desarrollo de prisma recto y pirámides.
13. Determinación de las trazas de la recta.

A todas las aplicaciones obtenidas de las tareas realizadas se les incorporó un instalador y una ayuda en línea, elementos que contribuyen a aumentar el nivel profesional de los productos obtenidos.

## **Conclusiones**

Se ha comprobado que, a través de la tarea extraclase, se pueden integrar varios contenidos de las disciplinas Programación, Matemática e Ingeniería de Software, así como de las asignaturas: Geometría Descriptiva, Dibujo, Gráfica por Computadoras y Diseño Asistido por Computadoras.

Integrando y preparando los recursos de años precedentes, así como los trabajos que deben realizar los estudiantes en su curso académico, propicia iniciar un trabajo que posteriormente permitirá obtener productos informáticos de calidad. Éstos pueden ser un gran aporte a la enseñanza de otras disciplinas que lo necesiten.

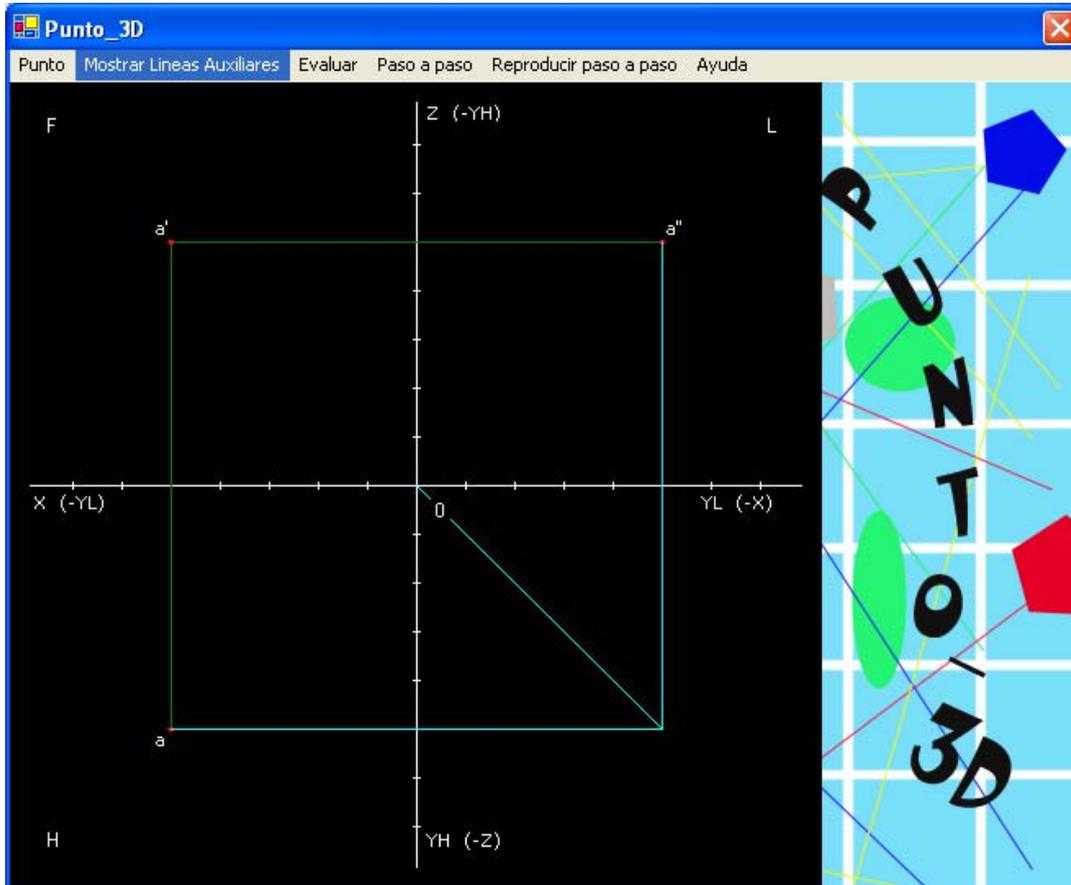
La tarea extractase, cuando es integradora, reduce los esfuerzos de los estudiantes y se puede planificar de forma que resuelva problemas prácticos, es un recurso educativo, que unido al proyecto, es de gran importancia en el desarrollo de las habilidades de los Ingenieros Informáticos y una vía importante para lograr la interdisciplinaridad.

## **Bibliografía**

1. Fernández, A., Pérez, R. "La Computación aplicada a la Física una opción para la interdisciplinariedad. Memorias del evento internacional DidacFisu'2002. SBN 959-16-0136-0. Matanzas. 2002.
2. Fernández, A., Pérez, R, et al. Gráfica por Computadora: aplicación de la POO, estructuras de datos y ficheros. Memorias del evento internacional Informática 2002. ISBN 959-237-079-6. La Habana. 2002
3. Foley, J. D., "Computer Graphics Principles and Practice", Addison–Wesley Publishing. 1997.
4. Iznaga, A. y I. Mallea, "Conferencias sobre Computación Gráfica", Facultad de Mecánica. ISPJAE. La Habana. 2002.
5. Pérez, R., Fernández, A., et al. Programación Orientada a Objetos aplicada a la Física, una opción para la interdisciplinariedad. Memorias del evento internacional Informática 2003. ISBN: 959-237-095-8. La Habana. 2003.

## Anexos.

Una de las aplicaciones realizada por los estudiantes para la representación de puntos en los 8 octantes, en una proyección ortogonal.



Cortesía de Lisandro Miranda Santana. Tarea de Gráfica por Computadora.