

TRABAJO PRACTICO INTEGRADOR: C++ Avanzado**A. Elementos necesarios**

TPs previos. Aspectos avanzados del lenguaje C++. Librerías del framework Qt5.

Anexo de “Referencias Básicas”.

B. Entrega y Evaluación

La fecha de entrega del informe y presentación es indicada al principio del ciclo, pero puede ser ajustada por mutuo acuerdo en clase.

Esta presentación es de modalidad presencial de el/los autor/es del trabajo, con exposición pública y posterior coloquio, abordando aspectos teórico-prácticos sobre la implementación realizada.

C. Actividades Generales

1. A partir del enunciado de la consigna específica, y de los temas especiales involucrados (Gráficos, Sonido, Interacción con Teclado y/o Mouse, Threads, Comunicaciones mediante protocolos y/o mecanismos de Red bajo arquitectura Cliente/Servidor, etc.), investigue sobre las posibilidades del lenguaje C++ para manejar los mismos bajo plataforma Linux¹.
2. Acuerde con el profesor los detalles que no estén especificados con claridad en este enunciado.
3. Proponga e implemente para el mismo, una Solución Orientada a Objetos en lenguaje C++ aplicando los conceptos fundamentales del ciclo: Clases, Objetos, Instanciación, Herencia, Agregación y Colecciones de Objetos, Polimorfismo, Streams de E/S, Constructores/Destructores, Sobrecarga de Métodos, Sobrecarga de Operadores, Plantillas.
4. Realice un informe bajo un formato similar el usado para el TP anterior, para el aplicativo resultante, cubriendo además los aspectos teóricos que estime apropiados sobre los temas que haya incluido y correspondan al ítem 1. Considere para ello el anexo “Esquema del informe”.
5. Elabore un video (no mayor a 15 minutos de duración) donde: en los primeros 2/3 del tiempo (aproximadamente) explique sintéticamente su solución, preferentemente basada en los gráficos y/o esquemas asociados tanto conceptuales como de la implementación realizada, y en el tercio restante del tiempo, demuestre al menos 2 ejecuciones diferentes y completas de su aplicación.
6. Como en los trabajos anteriores, envíe por correo al profesor los archivos resultantes (en este caso: código fuente del proyecto, incluyendo librerías de terceros usadas, ejecutable, informe y video).
7. Eventualmente, deberá realizar una ampliación o defensa grupal del trabajo realizado.

¹ Respecto de usar Windows como plataforma de desarrollo, hay diferencias de compatibilidad. Por ejemplo, con Qt5 no hay problema, mientras que con protocolo RPC no hay garantías (si bien hay herramientas tipo rpcgen desde la librería de wingnu32).

D. Consigna específica

1. Dominio del problema
 - Robótica.
2. Objetivo
 - Controlar un robot de 3 grados de libertad con efector final.
3. Requerimientos generales
 - El aplicativo/sistema debe estar representado e implementado usando OO.
 - Debe diseñarse usando modelos de capas, de manera que la solución pueda usarse independientemente de que haya interfaz gráfica o de consola.
 - El lenguaje principal debe ser C++ (aunque se admite interacción con Python).
 - La plataforma de explotación (producción) responde al sistema operativo Linux
 - Interfaces de usuario en modo gráfico con interacción estándar mediante mouse y teclado, preferentemente desarrolladas usando las librerías del framework Qt5.
 - La interfaz principal debe mostrar:
 - 3 vistas, correspondientes a cada plano de proyección ortogonal o bien, una vista en 3D del robot asignado. En cualquiera de los casos usar una estructura simplificada del robot, ejes/planos y escala (elegidos con criterio).
 - Panel de control del robot, que permita controlar:
 - Encendido/Apagado del robot. En el primer caso, el robot debe moverse a su “posición de origen”.
 - El movimiento, de cada uno de los vínculos (de manera independiente) especificando la velocidad, distancia o ángulo a recorrer y el sentido.
 - La actividad del efector final (propias de la herramienta particular elegida) especificando velocidad, tiempo de operación y sentido de operación (si corresponde).
 - Modo de trabajo del robot (manual o de aprendizaje y automático o de repetición), con selección del archivo de órdenes a ejecutar cuando corresponda.
 - Entrada directa de órdenes G-code.
 - Reporte, a intervalos regulares de tiempo o al inicio de cada orden, de las coordenadas sobre la trayectoria seguida para cada uno de los vínculos de interés del robot específico, con el instante de tiempo en que se alcanza la misma.
 - Curvas que muestren la velocidad de cada uno de los vínculos.
 - Se deben almacenar las órdenes de trabajo dadas desde el panel de control en archivos de texto, y también recuperarlas para su repetición. Cada una de las órdenes debe satisfacer el formato G-code estándar de RepRap.
 - El efector final no agrega grados de libertad especiales, más allá de algún desplazamiento fijo en las coordenadas asumidas para el punto de operación.
 - Posibilidad de consultar datos referidos al estado de operación del robot, estado de actividad del efector y parámetros de cada movimiento desde una aplicación remota en modo texto usando mecanismos RPC.

- Resulta deseable:
 - contar con efectos de sonido que indiquen el arranque/parada del robot, los cambios de posición del efector final y los momentos de actividad del mismo.
 - Generación de alarmas visuales/auditivas cuando se intenten desplazamientos fuera del espacio de trabajo (aproximado).
 - El aplicativo debe operar de manera completa con control desacoplado (aunque resulta mejor valorada aquella propuesta que avance en el control acoplado).
 - Definir constantes apropiadas para los parámetros extremos (límites de trabajo).
4. Robot específico
- Tipo: según anexo Configuraciones, asignado en clase
 - Dimensiones (extremas): anchura=600mm, altura=400mm, profundidad=400mm
 - Velocidades máximas: lineal=50mm/s, angular=1rad/s

E. Anexo: Referencias Básicas

1. Gráficos, mouse, teclado, sonido

- <https://usamossoftwarelibre.wordpress.com/2011/04/28/decide-que-libreria-grafica-usar-wxwidgets-vs-gtk-vs-qt/>

Con Qt

- https://es.wikibooks.org/wiki/Programaci%C3%B3n_con_Qt4
- <https://www.solvetic.com/tutoriales/article/1526-programaci%C3%B3n-multiplataforma-en-c-y-qt-bajo-entorno-gnulinux/>
- Videos de Curso
<https://www.youtube.com/watch?v=8FytWxfr2Is&list=PLkMd51xTELZLS4NQI71uOPvMPRveD9C1o>
- Video QtCreator <https://youtu.be/WSk8ojnCrrI>
- Video QtCreator (despliegue) <https://youtu.be/zHXdgPIF8Nc>

2. RPC:

- <https://github.com/Innova4DLab/DistributedSystems/tree/master/RPCGen>
- <http://www.chuidiang.org/clinux/rpc/rpc.php>
 - Video <https://www.youtube.com/watch?v=fBZPpJFaVFA>
 - Video http://youtu.be/pDJm_TQPhhE
- <http://dariolp.blogspot.com/2012/06/servidor-de-archivos-concurrente-rpc.html>
 - Código <https://code.google.com/archive/p/servidor-archivos-concurrente/>
- https://www.tamps.cinvestav.mx/~vjsosa/clases/sd/RPC_notas.pdf
 - Código https://www.tamps.cinvestav.mx/~vjsosa/clases/sd/rpc_prog.rar

3. G-code:

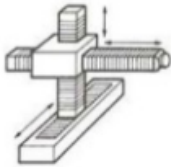
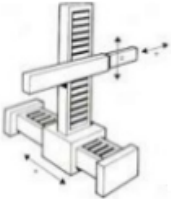

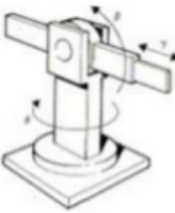
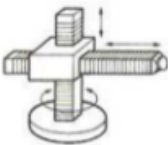
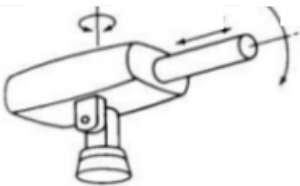

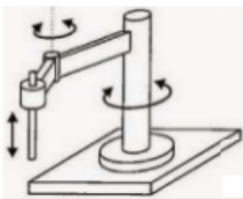
- <https://reprap.org/wiki/G-code/es>

F. Anexo: Esquema² del Informe

1. Carátula
 - a. Identificación de materia
 - b. Identificación del trabajo
 - c. Identificación de alumno/s
 - d. Identificación de docente/s
 - e. Identificación de carrera
 - f. Identificación de facultad/universidad
 - g. Identificación de ciclo lectivo
2. Introducción
 - a. Descripción breve del problema
 - b. Descripción detallada, con imágenes de robots del tipo asignado y ejemplos de implementación real si las hubiere.
 - c. Identificación (listado) de los Temas Especiales de C++ investigados
3. Marco teórico (desarrollo de cada uno de los Temas Especiales)
 - a. Conceptos/fundamentos
 - b. Esquema o gráfica descriptiva con detalles de aspectos estructurales o funcionales
 - c. Descripción de componentes software (librerías, frameworks, aplicaciones, etc.) asociados específicamente al problema/solución.
 - d. Descripción/ejemplificación paso a paso del modo de instalación/vinculación/uso, aplicado a la solución específica
4. Desarrollo
 - a. Diagrama de clases de la solución propuesta (debe ser consistente con la implementación)
 - b. Diagrama de secuencia de la solución propuesta (debe ser consistente con la implementación)
 - c. Diagrama de actividad (a nivel de aplicación) de la solución propuesta (debe ser consistente con la implementación)
 - d. Screenshots mostrando la ejecución del aplicativo, para cada una de las interfaces de entrada/salida propuestas
5. Comentarios y Conclusiones
 - a. Comentarios sobre las dificultades enfrentadas y su resolución, justificaciones a la solución propuesta (de diseño, de implementación, de instalación, etc.)
 - b. Análisis de ventajas/desventajas de los componentes usadas
 - c. Posibles extensiones a la solución dada
6. Referencias
 - a. Bibliografía/Sitios web consultados
 - b. URLs a los recursos software (tutorial, video, ejemplo, herramienta, etc.) que se hayan usado.

² El formato propuesto es un acercamiento al usado en tesinas de grado, posgrado e investigación

G. Anexo: Configuraciones

	1. PPP		5. PPP
	2. RRR Similar modelo ABB IRB 460 o FANUC 410ic		6. RRP
	3. RPP		7. RPR
	4. RRR Similar a modelo Mitsubishi RV-2AJ		8. RRP