Desarrollo de una aplicación para generación de horarios semanales de clase y estudio para una carrera específica de la Facultad de Ingeniería UdeA.

Este parcial consiste básicamente en la creación de una pequeña app capaz de generar horarios semanales de clase de una carrera especifica (Inicialmente Ingeniería Electrónica), en la cual se encuentren los cursos de la misma, además de indicar horas de estudio adicionales al estudiante dependiendo de la cantidad de créditos y horas trabajadas con docente. No hay restricción de funciones adicionales que mejoren la capacidad de dicha app, pero si contamos con una restricción a la hora de programar con la nula utilización de librerías y clases adicionales (como string, vector, etc).

ClassPlannerPro.

Para darle un poco de originalidad, decidí apostar por ClassPlannerPro (Planificador de Clases Profesional) como nombre de esta app, ya que describe fielmente de que trata la aplicación, además de que hace referencia al lenguaje en el que está basado (**C**lass**P**lanner**P**ro -> **CPP**).

Toda la información de los cursos de cada pensum será extraída de un archivo ‘.txt’ el cual tiene siempre la siguiente estructura.

[Nombre de la Carrera] <Código> (0000)

Donde todo lo que este dentro de [] representa el nombre de la carrera, <> el código de identificación propia de la Universidad de Antioquia y () en un numero de 4 dígitos en los cuales:

El primer número significa la cantidad se Horas Teóricas semanales, el segundo número significa la cantidad de Horas Practicas semanales, el tercero la cantidad de Horas Teórico – Practicas semanales y cuarta la cantidad de créditos que vale el curso.

Para que se ejecute el programa debe primero validar internamente si los archivos ´.txt´ se encuentren disponibles, en caso de que no estén, lanzar error y no ejecutar la aplicación, esto como medida de seguridad para evitar errores durante la ejecución del programa. Todo se va a guardar en 3 arrays, uno tipo char donde se encontrarán todos los cursos, un segundo arrays de tipo int para guardar los códigos y el ultimo de tipo int para guardar las horas de cada curso y su valor de crédito. Toda esta acción será realizada por la función ‘PensumReading()’, la cual retorna un booleano, que permite o no la ejecución del programa dependiendo si se logró abrir el archivo. **Nota**: Actualmente esta limitado a solo lectura de los cursos, para futuras actualizaciones se espera agregar divisiones por semestre de las materia e inclusión de más carreras en un mismo archivo.

El programa principal pregunta sobre el nombre del Usuario con el objetivo de anexarlo cuando se guarde el horario Generado. Sobre que carrera esta cursando para desplegar la lista de cursos de dicho programa. Luego, por medio de índices que cursos desea matricular, con restricciones de la cantidad de materias por créditos. Ya en este punto se le pide al usuario si de que forma quiere generar su horario, ya sea de forma manual o automática, además de que el calendario estará generando un horario en horas que van desde las 6AM a 10PM, los Lunes a Sábados para las horas con acompañamiento docente y de Lunes a Domingo para las horas practicas libres con las materias señaladas por su código de identificación.

Si el usuario decidió irse por la opción de generación automática, el programa va a generar un horario con los cursos elegidos por el mismo, el usuario tendrá la opción de elegir de:

* Aceptar el horario generado. Hace que el programa genere un archivo con la gráfica del horario e información adicional.
* Volver a generar un nuevo horario.
* Salir al menú anterior.

Si el usuario decidió por generar su horario de forma manual, el programa dará horarios aleatorios para que el usuario elija el mas conveniente por cada materia, cuando finalice este proceso, el programa asignara de forma automática los horarios de estudio adicionales, generando el archivo con el horario y finalizando el programa.