**ANÁLISIS DEL PROBLEMA**

1. **Problema:** Diseñar una aplicación que permita almacenar a través de una estructura dinámica de datos, las carreras con sus respectivas materias, secciones, profesores ,alumnos y notas por cada sección. Aunado a lo anterior, el sistema debe arrojar las siguientes estadísticas y reportes:
   1. Cantidad de secciones.
   2. Listado de secciones y profesores ordenadas por carrera.
   3. Listado de alumnos por sección con sus notas parciales y finales.
   4. Listado de alumnos aplazados por sección.
   5. Datos del alumno con el mejor promedio por sección.
   6. Datos de alumno mejor promedio de todo.
2. **Datos suministrados por el usuario:**
   1. Agregar carrera: nombre.
   2. Agregar profesor: cédula, nombre, apellido, dirección y teléfono.
   3. Agregar materia: código y nombre.
   4. Agregar sección: número y nombre del profesor.
   5. Agregar alumno: cédula, nombre, apellido y dirección.
   6. Nota alumno: número (del 0 al 20).
   7. Profesores por carrera: elegir carrera.
   8. Alumnos por sección: elegir carrera, materia y sección.
   9. Alumnos aplazados por sección: elegir carrera, materia y sección.
   10. Mejor nota por sección: elegir carrera, materia y sección.
3. **Operaciones**
   1. Agregar carrera: se mostrarán mensajes de error en casos de campo(s) vacío(s) y nombre repetido (consulta con el archivo carrera.obj). Todos los datos solicitados se almacenan en el archivo carrera.obj.
   2. Agregar profesor: se mostrarán mensajes de error en casos de campo(s) vacío(s) y cédula repetida (consulta con el archivo profesor.obj). Todos los datos solicitados se almacenan en el archivo profesor.obj.
   3. Agregar materia de una carrera específica: se mostrarán mensajes de error en casos de campo(s) vacío(s), código repetido y nombre repetido (consulta con el archivo materia.obj). Todos los datos solicitados se almacenan en el archivo materia.obj.
   4. Agregar sección de una materia específica: se mostrarán mensajes de error en casos de campo(s) vacío(s) y número de sección repetido (consulta con el archivo seccion.obj). Todos los datos solicitados se almacenan en el archivo seccion.obj.
   5. Agregar alumno en una sección específica: en caso que la cédula coincida con algún alumno registrado en la data, se auto-llenarán el resto de los campos coincidentes con dicha cédula (consulta con el archivo alumno.obj) .Se mostrarán mensajes de error en casos de campo(s) vacío(s), si la cédula ingresada coincide con la de algún alumno inscrito en la materia que está vinculada esta sección (consulta con el archivo inscritos.obj) y si la cédula ingresada coincide con la de algún profesor (consulta con el archivo profesor.obj). La cédula, nombre, apellido y dirección se guardarán en el archivo alumno.obj (si la cédula coincide con algún alumno, se actualizarán el resto de los datos de ese alumno). La cédula del alumno, el código de la materia y el número de la sección se guardarán en el archivo inscritos.obj.
   6. Asignar nota a un alumno de una sección específica: se mostrarán mensajes de error en casos de campo(s) vacío(s). La nota asignada se actualizará en el registro de esa misma sección en el archivo inscritos.obj.
   7. Profesores por carrera: se solicita la elección de la carrera (despliegue de listado de carrera a través del archivo carrera.obj). Dependiendo de la elección, se mostrarán aquellos profesores que dictan clases en dicha carrera (consulta con los archivos profesor.obj, sección.obj y materia.obj).
   8. Alumnos por sección: se solicita la elección de la carrera (despliegue de listado de carrera a través del archivo carrera.obj). Dependiendo de la carrera a elegir, se cargarán las materias coincidentes con dicha elección (despliegue de listado de materias a través del archivo materia.obj). De igual forma cargarán las secciones coincidentes con la materia a elegir (despliegue de listado de secciones a través del archivo seccion.obj). Se mostrarán mensajes de error en caso de no elegir los 3 parámetros establecidos. Dependiendo de la elección, se mostrarán los datos de los alumnos coincidentes (consulta con los archivos alumno.obj e inscritos.obj).
   9. Alumnos aplazados por sección: se solicita la elección de la carrera (despliegue de listado de carrera a través del archivo carrera.obj). Dependiendo de la carrera a elegir, se cargarán las materias coincidentes con dicha elección (despliegue de listado de materias a través del archivo materia.obj). De igual forma cargarán las secciones coincidentes con la materia a elegir (despliegue de listado de secciones a través del archivo seccion.obj). Se mostrarán mensajes de error en caso de no elegir los 3 parámetros establecidos. Dependiendo de la elección, se mostrarán los datos de los alumnos coincidentes (consulta con los archivos alumno.obj e inscritos.obj), tomando en cuenta que cada coincidencia en el archivo inscritos.obj, la nota sea menor a 10.
   10. Mejor nota por sección: se solicita la elección de la carrera (despliegue de listado de carrera a través del archivo carrera.obj). Dependiendo de la carrera a elegir, se cargarán las materias coincidentes con dicha elección (despliegue de listado de materias a través del archivo materia.obj). De igual forma cargarán las secciones coincidentes con la materia a elegir (despliegue de listado de secciones a través del archivo seccion.obj). Se mostrarán mensajes de error en caso de no elegir los 3 parámetros establecidos. Dependiendo de la elección, se mostrarán los datos del alumno coincidente, tomando en cuenta que contenga la calificación más alta (consulta con los archivos alumno.obj e inscritos.obj).