

# Programación Móvil

Dr. Marco Aurelio Nuño Maganda

Universidad Politecnica de Victoria  
Ingeniería en Tecnologías de la Información  
Agosto – Diciembre 2023

*[mnunom@upv.edu.mx](mailto:mnunom@upv.edu.mx)*

4 de septiembre de 2023

- Doctor en Ciencias Computacionales por parte del INAOE (2009).
- Profesor de Tiempo Completo de la UPV desde 2009.
- Miembro del Sistema Nacional de Investigadores - Nivel Candidado (2014-2016), Nivel I (2020-2022), Nivel I (2023-2027)
- 17 tesis dirigidas a nivel maestría.
- Asignaturas impartidas en el pasado
  - Licenciatura: Cómputo en Dispositivos Mviles, Graficación por Computadora Avanzada, Lenguajes y Automátas, Programación Orientada a Objetos
  - Maestría: Visión por computadora, Tópicos Selectos de Imagenología, Fundamentos de Sistemas de Información
- Miembro del Núcleo Académico Básico (NAB) de la maestria en Ingeniería de la UPV.

- Nombre de la clase: **Programación Móvil-Agosto – Diciembre 2023**
- Código de clase en Classroom: **zpj7hb7**
- Enlace Meet para sesiones no presenciales:  
**<https://meet.google.com/ezm-cukx-ehz>**

- Días y horas de clase

Clave de Grupo	Lunes	Martes	Miércoles	Jueves	Viernes
ITI-271086		12:05-13:00	14:00-14:55	12:05 - 13:55	12:05 - 13:55
ITI-271093	14:54-15:50		14:55-15:50	14:00 - 15:50	14:00 - 15:50

- Fechas Importantes:

- Inicio de Cursos: 4/Septiembre
- Fin de Cursos: 12/Diciembre
- Dias no hábiles oficiales: 20 de noviembre (lunes).

- Se recomienda puntualidad y asistencia a las sesiones.
- Respecto hacia el profesor y hacia sus compañeros y compañeras.
- No se permite el ingreso y/o ingestión de **Alimentos** ni **Bebidas** de ningún tipo a la clase (ni aún estando en sus hogares).
- **NO SE PERMITE EL USO de AUDIFONOS O DISPOSITIVOS MANO-LIBRES EN CLASE.** De detectar esta situación, se amonestará al estudiante y de reiterar, dicho estudiante será expulsado de **CURSO** por el resto del **CUATRIMESTRE**, sin derecho a réplica.

- Se pasa lista al inicio de la clase. En caso de reincorporación tardía, se pone un retardo.
- Tres RETARDOS equivalen a una INASISTENCIA, que no es JUSTIFICABLE.
- Para actividades en línea, debe utilizar su cuenta institucional de la universidad, ya que no se autorizará el ingreso de ningún participante externo a la universidad.
- Para justificar una inasistencia, es necesario cumplir con los siguientes pasos:
  - Agendar una asesoría de la clase mediante el SIITA, verificar que el profesor marque la asesoría como realizada y confirmar la asesoría.
  - En cualquier espacio al final de una clase, solicitar al profesor su registro de justificación.
  - **NO ES NECESARIO ENVIAR correo electrónico al profesor –**

# Asistencias mínimas para aprobar asignatura

- Al NO alcanzar un 80 % de asistencia, el estudiante pierde su derecho de ser EVALUADO

## Alumnos VIPs

En caso de tener un empleo formal dentro o fuera de la ciudad, es necesario entregar una **constancia laboral** que acredite el horario que se está cubriendo (en el caso de locales, este horario se debe empalmar con el de la materia). En esa constancia debe acreditar que se está haciendo labores de manera presencial en tal ubicación. Esto lo dispensa solo del requisito de las asistencias, mas no de los proyectos que deban entregarse. Incluso pudiera solicitarle presentar avance de manera “remota” durante alguna de las clases. Enviar esa constancia con copia para el director de carrera.

- Todo aquel estudiante que realice una estancia de investigación, visita académica, visita a empresa, evento deportivo (por parte de la Universidad), o cualquier otra actividad académica, debe negociar una extesión máxima de **7 días naturales** de prórroga para entrega de proyectos individuales, en equipo o asignaciones especiales, respaldado por algun documento oficial que acredite dicha actividad.
- Ninguna actividad de las anteriormente mencionadas **es causa de EXENSIÓN** de los proyectos solicitados en la clase, ni generará ningún cambio o ajuste a sus calificaciones por participar en dicho evento. De no cumplir con los plazos de entrega establecidos, deberá asumir las consecuencias de no entregar los proyectos o trabajo solicitado en el tiempo establecido.
- Las inasistencias a clase quedan justificadas solo **durante el desarrollo del evento y los traslados**. Para justificar dichas inasistencias, debe hacerlo mediante el procedimiento establecido para dicha justificación.



- ① Introducción al cómputo móvil
  - ① Fundamentos de Programación móvil
  - ② Tipos de datos y expresiones
  - ③ Entornos de desarrollo de aplicaciones móviles
  - ④ Estructura de proyectos móviles
- ② Diseño de Aplicaciones Móviles
  - ① Interfaz de usuario
  - ② Desarrollo de Aplicaciones Móviles
  - ③ Servicios y Notificaciones en Aplicaciones Móviles
- ③ Empleo de sensores en dispositivos móviles
  - ① Gestión de Sensores
  - ② Tópicos selectos de programación móvil

Para cada unidad del curso, se consideran 3 aspectos:

- Ejercicios o investigaciones especiales (1)- 25 %
- Proyecto Individual - 35 %
- Proyecto en Equipo - 40 %

Para aprobar el curso, es obligatorio tener calificación aprobatoria en todas las unidades. Para tener derecho a una evaluación de recuperación de la unidad, el estudiante debe haber cumplido con 2 /3 requisitos de la unidad, y esta recuperación aplica solamente UNA VEZ a lo largo del cuatrimestre.

Para cada unidad, habra sesiones de “teoria”, sesiones de seguimiento de proyectos y sesiones de esparcimiento

- En las sesiones de teoria, el profesor presentara uno o varios temas
- En las sesiones de seguimiento de proyectos, de manera aleatoria se nombrara al integrante de equipo individual o en equipo. En el caso de que un integrante individual no responda, se le bajarán 5 puntos a su calificación del proyecto
- En las sesiones de esparcimiento, se permitirá a los estudiantes trabajar en proyectos pendientes, pero se contabilizará la asistencia.

## Sesiones de Seguimiento de proyectos

- En el caso de que el integrante del equipo seleccionado aleatoriamente no responda satisfactoriamente lo cuestionado, se le bajaran 5 puntos a su calificación del proyecto a todos los integrantes del equipo
- En el caso de los proyectos en equipo, el integrante seleccionado es aleatorio. Si en una primera ronda le toco al integrante A, en una segunda ronda posiblemente le toque al integrante B

Lo que se debe presentar en una sesion de seguimiento de proyectos

- En un trabajo individual
  - Compartir pantalla de la ejecucion del avance del proyecto
  - Explicar con recursos multimedia los pasos para la resolucion del proyecto
  - Establecer el avance desde la ultima entrega
- En un trabajo grupal
  - Compartir pantalla de la ejecucion del avance del proyecto
  - Explicar con recursos multimedia los pasos para la resolucion del proyecto
  - Desglosar como se repartio el trabajo entre los integrantes del equipo
  - Establecer el avance desde la ultima entrega

Acerca de los proyectos

- Aleatorios y DIFERENTES para la mayoría (preferentemente para cada integrante)
- Equipos: Proyectos diferentes para cada equipo, e Integrantes de los mismos formados de manera ALEATORIA!!

## Acerca de la participacion

- Se selecciona al azar un estudiante, existen varias posibilidades:
  - Esta presente, puede abrir camara, microfono y compartir escritorio para responder lo solicitado, y responde → POSITIVA
  - Esta presente, NO puede abrir camara, ni microfono, y no comparte escritorio lo solicitado, no es necesario que responda → NEGATIVA
  - Esta presente, pero decide no participar → NEGATIVA
  - NO esta presente → NEGATIVA
- La calificacion de Participación es proporcional al porcentaje de participaciones positivas (7/10, 3/5, 0/4, 3/3).

# Cartucho de Recuperación (REC)

- Estudiante tiene derecho a solicitar un ÚNICO proyecto de recuperación para las primeras dos unidades del curso.
- Esta solicitud debe HACERLA es estudiante - El profesor NO ES RESPONSABLE de informar al estudiante cuando tiene un ADEUDO.
  - SOLO es posible Recuperar un Proyecto (Individual o en Equipo) por CUATRIMESTRE
- Si el proyecto no entregado es individual, se asigna otro proyecto diferente.
- Si el proyecto es en equipo, de común acuerdo con los integrantes pueden trabajar en otro proyecto diferente en equipo, o recibir una asignación individual de un proyecto diferente.
- Por cuestion de tiempo, NO SE APLICAR RECUPERACION de UNIDAD 3.
- La calificación recuperada será asignada siempre y cuando cumpla con el porcentaje de falta mínimo necesario para aprobar.

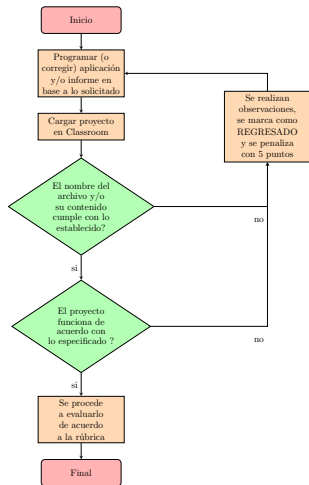


- Para cada práctica realizada, entregar un documento (**únicamente en formato PDF\***) con las siguientes secciones:
  - Introducción
  - Desarrollo Experimental
  - Resultados
  - Conclusiones
  - **Referencias**
- Para GENERAR este reporte es necesario utilizar la plantilla en LATEX (**únicamente usando LATEX\***) localizada en el siguiente enlace:  
<https://www.overleaf.com/read/dgkhvfwynygvc>

- Bajo ninguna circunstancia deben incluir **CÓDIGO FUENTE**. Si pueden incluir diagrama de flujo, Pseudocódigo, Diagrama E-R, Diagrama de Clases, de Casos de USO, etc. De incluir código fuente, solo tendrá un 50 % del valor en la calificación.
- En caso de trabajos individuales o en EQUIPO, deben emplear la plantilla LaTeX que se provee. En caso de utilizar algo diferente a LaTeX u otra plantilla de LaTeX, la calificación proporcional del informe será **DESESTIMADA**.
- En caso de trabajos en equipo, se debe agregar los integrantes al inicio del INFORME. **El trabajo solo cuenta para aquellos integrantes mencionados en el informe (y que dicho nombre se encuentre registrado tal cual en la lista). Una vez ENTREGADO, si hay OMISIONES de los integrantes, no se realizará CORRECCION alguna, se debe asumir la consecuencias que esto conlleva.**

- Proyecto: 66 Puntos
  - Ejecución y Funcionalidad: 45 Puntos
  - Modularidad: 13 Puntos
  - Documentación: 8 Puntos
- Informe: 34 Puntos
  - Uso adecuado de Latex: 5 Puntos
  - Organización y Redacción: 6 Puntos
  - Referencias en formato adecuado: 8 Puntos
  - Evidencia del trabajo realizado: 8 Puntos
  - Sin faltas de ortografía ni errores de dedo: 7 Puntos

- Proyecto que no este entregado de acuerdo con las especificaciones, será regresado (y penalizado)
- Puede dar tantas vueltas como el estudiante desee
- Se recomienda LEER con cuidado la sección de entregables en esta diapositiva



En cada entrega, subir un archivo .ZIP, cuyo nombre de archivo debe seguir la siguiente especificación (todo en minúsculas , sin ESPACIOS):

- **Clave de GRUPO (incluir guión medio)**
- **Nombre del integrante iniciando por apellido paterno, SIN ESPACIOS y separado por guion bajo**
- *clavegrupo\_nuno\_maganda\_marco\_aurelio.zip*,  
*clavegrupo\_nuno\_maganda\_marco\_aurelio.pdf*, etc.
- El contenido de dicho archivo debe ser el siguiente:
  - 1 Archivo .ZIP con el código fuente.
  - 2 Archivo instalador de la aplicación (.APK).
  - 3 Archivo PDF con el informe.
- Mismo formato de NOMBRE de archivo del ENTREGABLE principal para el nombre de los archivos al interior del ZIP

- Este es un ejemplo, pero nunca falta el listo que lo entrega tal cual!
- **Archivo Principal:** iti-271086\_nuno\_maganda\_marco\_aurelio.zip
- Contenido de dicho archivo:
  - **iti-271086\_nuno\_maganda\_marco\_aurelio.zip** (Codigo fuente)
  - **iti-271086\_nuno\_maganda\_marco\_aurelio.apk** (Instalable)
  - **iti-271086\_nuno\_maganda\_marco\_aurelio.pdf** (Informe)

**Cuatro cuatrimestres ignorando estas instrucciones, ya debería poner nombres y apellidos. El que lo haga serán sentadillas o páginas, ustedes escojan!**

En cada entrega, **UN SOLO INTEGRANTE DEL EQUIPO** deberá subir un archivo .ZIP, cuyo nombre de archivo debe seguir la siguiente especificación (todo en minúsculas):

- **Clave de GRUPO (incluir guión)**
- **Palabra equipo seguido del numero de equipo (usando dos digitos)**
- Por ejemplo: **iti-271086\_equipo\_01.zip**

El contenido del archivo debe serL

- **iti-271086\_equipo\_01.zip** (Codigo fuente)
- **iti-271086\_equipo\_01.apk** (Instalable)
- **iti-271086\_equipo\_01.pdf** (Informe)

En el caso de nombres y apellidos acentuados, con diéresis o con virgulilla (~), sustituir de acuerdo con las siguientes reglas:

- Sustituir N/n por Ñ/ñ
- Sustituir A/a por Á/á
- Sustituir E/e por É/é
- Sustituir I/i por Í/í
- Sustituir O/o por Ó/ó
- Sustituir U/u por Ú/ú
- Sustituir U/u por Ü/ü



- Se debe extraer utilizando el SCRIPT para dicho propósito (SOLO EXISTE HAY PARA LINUX, usuarios Windows deben programar el SUYO o replicar los pasos del SCRIPT Linux para su entrega) que se proveerá en el CLASSROOM para extraer los archivos necesarios
- **De ninguna manera se debe compactar la carpeta completa del PROYECTO. De hacer esto, habrá una penalización de 20 PUNTOS**

# Fechas importantes de entrega de proyectos

- Fecha de asignación: fecha en que se da a conocer al grupo el trabajo a elaborar
- Fecha de entrega sin penalización: 14 días naturales después de la fecha de asignación
- Fecha de entrega máxima: 5 días naturales después de la fecha de asignación, después de esa fecha el proyecto tendrá valor CERO aun si es entregado en la plataforma
- Penalización por entrega tardía: 15 puntos.
- Penalización por entrega posterior al plazo máximo: 25 puntos (Acumulable con la penalización anterior).

## LINUX

### Recomendaciones

- No es obligatorio instalarlo, pero es recomendable por cuestión de desempeño.
- Si no quieren formatear computadora, se recomienda utilizar un HD booteable (SSD con persistencia) y bootear desde su laptop o computadora.
- Si lo instalan de manera nativa, puede ser cualquier distribución (**Mint, Ubuntu, Lubuntu, Xubuntu, Debian**).

### Dispositivo Físico con Android

- Teléfono Inteligente/Tablet con Android Instalado (No afecta si no es la última versión)

Sobre una instalación de Linux, se debe instalar lo siguiente:

- Android Studio
- Scrcpy (<https://github.com/Genymobile/scrcpy>)
- Navegador Chrome/Firefox actualizado
- LaTeX para edición de reportes

Los lenguajes soportados por Android Studio son:

- Kotlin
- Java

Se hará énfasis en el lenguaje Kotlin, dado que es el lenguaje recomendado para nuevas aplicaciones, sin embargo, podría trabajarse también en Java dependiendo del proyecto

Se buscan integrantes para ingresar al  
Salon de la fama del PLAGIO

- Reprobación automática a quien reproduzca códigos de otros compañeros y los reporte como suyos, además de una nota en su expediente con copia para el consejo de calidad



- Reprobación automática a quien copie códigos de Internet y los reporte como suyos, además de una nota en su expediente con copia para el consejo de calidad



```

1 //Función que calcula el número de píxeles de cada nivel de intensidad
2 int contarPixelas(const Image& img) {
3     int n = 0;
4     for (int i = 0; i < img.getHeight(); i++) {
5         for (int j = 0; j < img.getWidth(); j++) {
6             n++;
7         }
8     }
9     return n;
10 }
11
12 //Función que calcula el histograma
13 void calcularHistograma(const Image& img, int* hist) {
14     int n = contarPixelas(img);
15     for (int i = 0; i < 256; i++) {
16         hist[i] = 0;
17     }
18     for (int i = 0; i < img.getHeight(); i++) {
19         for (int j = 0; j < img.getWidth(); j++) {
20             int pixel = img.getPixel(i, j);
21             hist[pixel]++;
22         }
23     }
24 }
25
26 //Función que calcula la función de distribución acumulada (CDF)
27 void calcularCDF(int* hist, int* cdf) {
28     int n = contarPixelas(img);
29     for (int i = 0; i < 256; i++) {
30         cdf[i] = 0;
31     }
32     for (int i = 0; i < 256; i++) {
33         cdf[i] = hist[i];
34         for (int j = i + 1; j < 256; j++) {
35             cdf[j] += hist[i];
36         }
37     }
38 }
39
40 //Función que normaliza el histograma
41 void normalizarHistograma(int* hist, int* histNorm) {
42     int n = contarPixelas(img);
43     for (int i = 0; i < 256; i++) {
44         histNorm[i] = (float) hist[i] / n;
45     }
46 }
47
48 //Función que muestra el histograma
49 void mostrarHistograma(const Image& img, const char* name) {
50     int n = contarPixelas(img);
51     int* hist = new int[256];
52     calcularHistograma(img, hist);
53     int* cdf = new int[256];
54     calcularCDF(hist, cdf);
55     int* histNorm = new int[256];
56     normalizarHistograma(hist, histNorm);
57     //Mostrar el histograma
58     int max = 0;
59     for (int i = 0; i < 256; i++) {
60         if (histNorm[i] > max) {
61             max = histNorm[i];
62         }
63     }
64     //Normalizar el histograma entre 0 y 255
65     for (int i = 0; i < 256; i++) {
66         histNorm[i] = (int) (histNorm[i] * 255 / max);
67     }
68 }

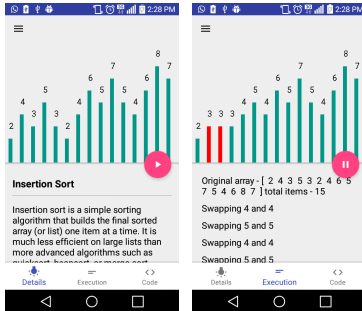
```



- Reprobación automática a quien copie códigos de Internet y los reporte como suyos, además de una nota en su expediente con copia para el consejo de calidad

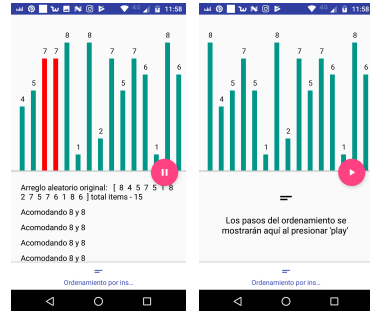


<https://github.com/naman14/AlgorithmVisualizer-Android>



Proyecto Original

Dr. Marco Aurelio Nuño Maganda



Proyecto "clonado"

Programación Móvil

“Finalmente son jóvenes que están en la preparatoria y que deben de leer su convocatoria con toda claridad, si no cumplen con los requisitos, si no pueden leer una convocatoria que dice tienes que traer número uno esto, número dos esto, número tres esto, no están listos para ser **estudiantes de educación superior**, así lo digo con toda claridad”.

Sara Ladrón de Guevara.

Rectora de la Universidad Veracruzana (2013-2017 y 2017-2021).

