



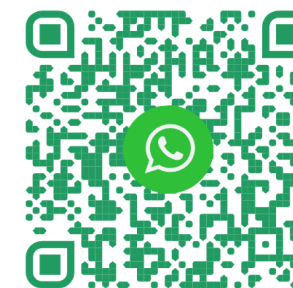
EXAMEN DE RECUPERACIÓN PROGRAMACIÓN I

(RRA20) SOFTWARE
ICCD144-PROGRAMACIÓN I-GR1SW

Patricio Michael Paccha Angamarca
Magister en ingeniería de software
Diplomado Superior De Cuarto Nivel En Gerencia Estratégica De Mercadeo
Ingeniero en Sistemas Informáticos Y Computación



CLASS ROOM



CLASS GROUP



AULA VIRTUAL



EXAMEN: 2do Bimestre



TIPO DE INSTRUMENTO: Examen – 2do Bimestre

PERIODO ACADÉMICO: 2022-B

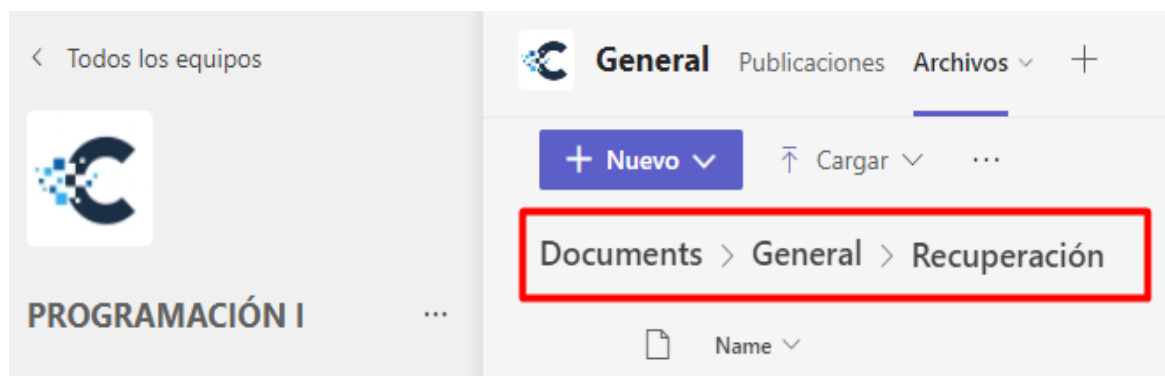
TIEMPO : 3 Horas (Inicio: 11:00 am - Fin: 13:00 pm)

PUNTAJE: 20 puntos

20.mar.2023

OBLIGATORIO :

- Leer cuidadosamente el caso práctico de la presente rúbrica.
- Crear la carpeta/directorio **apellido.nombre** e incluir la presente rúbrica en este directorio subido. Finalmente se debe subir lo desarrollado en el al **Teams** en:



- **Se calificará** únicamente los exámenes entregados dentro del tiempo establecido.
- Si se detecta copia de código se anulará su examen.



CASO PRACTICO:

CLASSIFIED

UCRANIA (Kiev)

Los países de la OTAN y la Unión Europea han proporcionado arsenal bélico de ultima generación a Ucrania, el cual se describe a continuación:

Código	Tipo Arsenal Bélico
a	Aeronaves
b	Bombarderos
c	Convoy
d	Drones
t	Taques



Ucrania inmediatamente establece la estrategia de ataque fijando coordenadas/posiciones en territorio para hacer frente a Russia.

THE RUSSIAN FEDERATION (Kremlin)

Rusia, ha decidido crear **bombas inteligentes** (BOMB-I#) basada en autómatas para reconocer el tipo de arsenal bélico ucraniano y determinar el tipo de bomba inteligente para un destrucción eficiente de la coordenada/posición ucraniana. Los tipo de bombas a construir son:

Tipo Bomba	Patrón (AFD)
BOMB-II	a^*b^+c
BOMB-IP	ab^*c^+
BOMB-IPi	a^+bc^*

El grupo de inteligencia del Kremlin hackea la estrategia de ataque ucraniano identificando las coordenadas/posiciones, capacidad bélica y tipo de arsenal bélico de cada coordenada.



CASO PRACTICO:

CLASSIFIED

Coordenadas
UCRANIA



RUSSIA



Descripción de la estrategia de ataque ucraniano por coordenada:

Cap = capacidad bélica
Geo = geoposicionamiento
Arsenal = tipo arsenal

Cap,	Geo,	Tipo Arsenal
0,	GPS0,	ab
1,	GPS1,	bc
2,	GPS2,	ac
3,	GPS3,	bcd
4,	GPS4,	acd
5,	GPS5,	bct
6,	GPS6,	act
7,	GPS7,	aaabbct
8,	GPS8,	abbccdt
9,	GPS9,	aaabbcd





EXAMEN: 2do Bimestre

INSTRUCCIONES:

- a) El examen/proyecto/solución debe estar contenido en el directorio **apellido.nombre** con la presente rúbrica y subilo al **TEAMS**
- b) Las **variables globales, variables locales, procedimientos, funciones y estructuras** deben usar el prefijo conformado por la primera letra/vocal de su nombre y apellido usando notación **camelCase**. Ejemplo si el alumno se llama: Pepe Lucho Álvarez Perez

```
int paNombre;  
void paGetEdad() {...}
```

- c) En el caso de constantes se deben crear todas en mayúsculas y si son palabras compuestas se debe usar guion bajo. Ejemplo:

```
const string PANOMBRE_COMPLETO = "Pepe Lucho Álvarez Perez";
```

- c) **OBLIGATORIO, el proyecto debe estar en el Github desde el inicio del examen. Cada horas deben haber commits. Al finalizar el examen agregar la url del github en el README.md**

- d) **Los procedimientos, funciones y estructuras - código debe estar documentado**

- e) Crear al menos una **librería** cumpliendo con los ítems b, c, d para colores, validaciones, etc. Ejemplo si el alumno se llama:

Pepe Lucho Pérez Suarez

nombre de librería: ./lib/**ppColor.h**
./lib/**ppUtil.h**

NOTA:

Si la aplicación tiene algún tipo de **crash**:
No cumplir con estas instrucciones :

-0,5 puntos por crash
-0,5 puntos por ítem





EXAMEN:

Desde el Kremlin(Rusia) te han contratado para que desarrolles un programa (c/c++) que cumpla con los siguientes requisitos:

1. Conforme los **ítems b, c, d** declarar / inicializar/ definir :

- Definir constantes globales con sus datos: **Cedula y Nombre Completo con valores todo en mayúsculas**
- Crear estructura **coordenada** { capacidad bélica (entero), geolocalización, detalle del Arsenal (Cadena de caracteres) y punteros que permitan estructurar un árbol binario de búsqueda}
- Crear un **archivo de texto** que contenga las coordenada ucranianas (Cap, Geo, Tipo Arsenal) cabeceras y datos. Cada fila corresponde a su número de cedula en inverso. Ejemplo: 1103635449

3 PUNTOS

Cap,	Geo,	Tipo Arsenal
9,	GPS9,	aaabbcd
4,	GPS4,	acd
4,	GPS4,	acd
5,	GPS5,	bct
3,	GPS3,	bcd
6,	GPS6,	act
3,	GPS3,	bcd
0,	GPS0,	ab
1,	GPS1,	bc
1,	GPS1,	bc





EXAMEN:



2. Crear un procedimiento para leer el archivo de texto que contiene las coordenadas y cumplir con:

- Cada línea que se lee debe mostrar un **loading** de carga que va de 0% a 100% y luego se muestra la línea leída del archivo
- Cada línea a leer debe ser una coordenada, caso contrario es un errores a mostrarse en color rojo y debe mostrar el tipo de error. El error no debe bloquear la ejecución del programa y debe avanzar la siguiente línea. Ejemplo al leer la primera línea del archivo el encabezado no es una coordenada por lo cual se debe mostrar un error
- Cada coordenada (Cap, Geo, Tipo Arsenal) leída correctamente se deben presentar en color verde y se debe agregar a una **PILA**. Cada vez que se agrega una coordenada debe realizarse con el procedimiento **PUSH** y mostrar esta acción en la consola (PUSH).

Ejemplo:

```
< skull > ./"examen3.exe"
[+]APILA coordenadas ...
>>(ERROR) Cap,Geo,      tipoArsenal -> stoi
(PUSH) 100% 9, GPS9, aaabbcd
(PUSH) 100% 4, GPS4, acd
(PUSH) 100% 4, GPS4, acd
(PUSH) 100% 6, GPS6, act
(PUSH) | 22%
```





EXAMEN:



3. Presentar los datos del desarrollador y **PILA** de la capacidad bélica Ucrania, con los siguientes ítems :

4 PUNTOS

- **Developer-Nombre**, usar procedimiento para presentar el nombre en letra capital. (primera letra de cada nombre en mayúsculas y el resto de letras en minúsculas)
- **Developer-Cedula**, número de cedula del desarrollador
- **Capacidad Bélica** que es la suma de la capacidad bélica (Cap) de cada coordenada obedece a cada PUSH en la PILA
- **Coordenada-Total**, cantidad de push (PUSH) realizados a la pila

Ejemplo:

```
[+]PILA : Capacidad belica Ucrania
Developer-Nombre : Pepe Lucho Perez Suarez
Developer-Cedula : 0123436445
Capacidad Belica : 52
Coordenada-Total : 12
```





EXAMEN:

4. Crear un **diagrama del autómata determinista finito - ADF** para un tipo de BOMBA.

Considere su último número de cédula, como se indica a continuación:

Tipo Bomba	Patrón Exp. Reg. (AFD)	Último Nro. Cédula	Agregar alfabeto a su Patrón Exp. Reg. (AFD)
BOMB-II	a^*b^+c	impar	d^+ ó t^+ , obligatorio agregar drones ó tanques
BOMB-IP	ab^*c^+	par	d ó t , obligatorio agregar dron y tanque
BOMB-IPI	a^+bc^*	cero	d^* ó t^* , obligatorio agregar drones ó tanques

4. Conjuntamente al diagrama agregar la descripción en lenguaje de conjuntos:

- Estados de su autómata $Q=\{\dots\}$
- Estados Finales de su autómata $F=\{\dots\}$
- Matriz de transición según el **diagrama del autómata determinista finito - ADF** realizado.

5. **DESAPILAR** todas las coordenadas de la **PILA** mediante el procedimiento **POP**.

- Las coordenadas debe tener color
- Cada POP que se realice debe mostrar la estructura: (POP) **Geo.Cap** { **tipoArsenal** BOMBA(si aplica al tipoArsenal y mostrar en color rojo) } .
- Cuando la pila queda vacía presentar "<< PILA VACIA >> "
- Guardar cada (POP) en un archivo de texto (nombre del archivo = apellidoNombreCoord.txt). Debe ser igual a su salida de consola
- Guardar cada (POP) en un archivo de texto **solo de las coordenada que aplica bomba** (nombre del archivo = apellidoNombreCoordBomba.txt)

```
[+]DESPILA COORDENADAS & BOMBA
(POP) GPS8.8 = { abbccdt }
(POP) GPS7.7 = { aaabbct }
(POP) GPS2.2 = { ac }
(POP) GPS1.1 = { bc BOMBIA_II }
(POP) GPS0.0 = { ab BOMBIA_II }
(POP) GPS3.3 = { bcd }
(POP) GPS5.5 = { bct }
(POP) GPS3.3 = { bcd }
(POP) GPS6.6 = { act }
(POP) GPS4.4 = { acd }
(POP) GPS4.4 = { acd }
(POP) GPS9.9 = { aaabbcd }
<< PILA VACIA >>
```





La tradición y el prestigio de la Politécnica exigen que el comportamiento de sus miembros se encuadre en el respeto mutuo, la honestidad, el apego a la verdad y el compromiso con la institución.

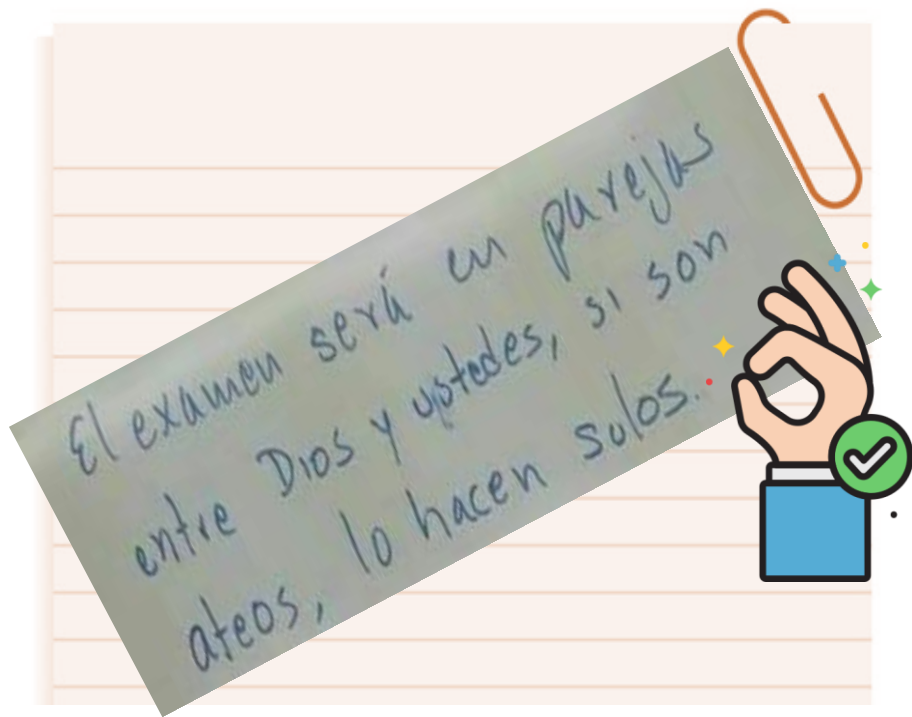
Con tal antecedente, el presente Código de Ética define la norma de conducta de los miembros de la Escuela Politécnica Nacional:

RESPECTO HACIA SÍ MISMO Y HACIA LOS DEMÁS

- Fomentar la solidaridad entre los miembros de la comunidad.
- Comportarse de manera recta, que afirme la autoestima y contribuya al prestigio institucional, que sea ejemplo y referente para los demás.
- Respetar a los demás y en particular la honra ajena y rechazar todo tipo de acusaciones o denuncias infundadas
- Respetar el pensamiento, visión y criterio ajenos.
- Excluir toda forma de violencia y actitudes discriminatorias.
- Apoyar un ambiente pluralista y respetuoso de las diferencias.
- Convertir la puntualidad en norma de conducta
- Evitar el consumo de bebidas alcohólicas, tabaco, sustancias psicotrópicas o estupefacientes.

HONESTIDAD + VERDAD + COMPROMISO CON LA INSTITUCIÓN



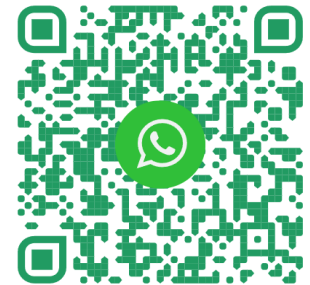


PROGRAMACIÓN I
(RRA20) SOFTWARE
ICCD144-PROGRAMACIÓN I-GR1SW

Patricio Michael Paccha Angamarca
Magister en ingeniería de software
Diplomado Superior De Cuarto Nivel En Gerencia Estratégica De Mercadeo
Ingeniero en Sistemas Informáticos Y Computación



CLASS ROOM



CLASS GROUP



AULA VIRTUAL