

Examen sobre OpenMP C/C++
(Duración: 2 horas)

Instrucciones:

1. Puede usar el IDE que prefiera, se recomienda trabajar con VS-2019.
2. A través del punto de entrega en el sitio del curso, “suba” su trabajo en un archivo cuyo nombre incluya su número de carnet: B67890.cpp.
3. GUARDE SU TRABAJO CADA 5 O 10 MINUTOS PARA MINIMIZAR PÉRDIDAS EN CASO DE FALLA DEL FLUIDO ELÉCTRICO.
4. NINGÚN CÓDIGO COMENTADO SERÁ TOMADO EN CUENTA PARA LA EVALUACIÓN.
5. Su programa será evaluado con base en tres criterios: **eficiencia**, **eficacia** y **simplicidad**.
6. Cualquier evidencia de fraude será penalizada fuertemente con base en el reglamento de régimen estudiantil.

Para el famoso juego “busca minas” (Ver: <https://cardgames.io/minesweeper/>), se puede representar el campo minado mediante tres matrices NxN:

1. el “campo real” CR,
2. el “campo visible: CV,
3. el riesgo: R.

En CR cada casilla toma un valor del conjunto {0, 1, 2...8, 10}, donde un valor entre 0 y 8 indica cuántas minas hay en casillas adyacentes y 10 indica que hay una mina en la casilla.

En CV cada casilla toma un valor del conjunto {-1, 0, 1, ... 8}, donde -1 indicaría que es una casilla oculta, y un valor entre 0 y 8, la cantidad de minas adyacentes.

En R cada casilla toma un valor en el intervalo real [-1, 0] para indicar en cada casilla cerrada el riesgo de abrirla y encontrar una mina.

El campo minado tiene forma toroide, por lo que TODAS las casillas tienen ocho adyacentes. Bajo este contexto, desarrolle los ejercicios siguientes usando OpenMP / C++ :

#1 Con base en la especificación y el código provisto programe la función “int genera_minas(double p, T_campo& campo)”.

#2 Con base en la especificación y el código provisto programe la función “void cuenta_minas(T_campo& campo)”.

#3 Con base en la especificación y el código provisto programe la función “void estima_riesgo_campo(T_campo& campo_visible)”.

Buena suerte!!!!