

- 1) Ici chaque thread se charge d'une cellule, chaque cellule va vérifier ses 8 voisins
- 2) Sans optimisation, chaque cellule va procéder à la lecture de chacun de ses voisins, c'est-à-dire 8 lectures + la lecture de sa propre valeur nous pouvons donc dire que chaque cellule fait 9 lectures. Ces 9 lectures ne sont pas coalescente.
- 3) Les threads vont lire trois emplacements mémoire consécutifs correspondant aux voisins supérieurs, puis effectuer un saut pour lire les trois cases ( dont la sienne ) correspondant à la ligne où se trouve la cellule, puis un nouveau saut pour lire les trois cellules de la lignes des voisins inférieurs.
- 4)
- 5) Le tableau de cellules est maintenant partagés par la ligne "extern \_\_shared\_\_ int cells[;]"
- 6) La divergence se fait entre le moment où on lis une cellule et le moment où l'on va la modifier. On utilise la fonction \_\_syncthreads pour attendre que tous les threads aient eu le temps d'écrire la valeur de leur cellule dans le tableau de cellules partagé.
- 7) Lorsque l'on compile avec l'option de compilation -Xptxas=-v, nvcc nous affiche 27 registres.
- 8)
- 9)