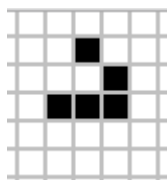
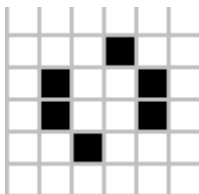
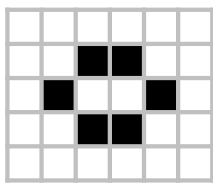


## Introdução a Computação

### Autômatos celulares 2D

- 1) Explique as principais diferenças entre autômatos celulares 1D e 2D em termos da topologia da rede e vizinhança. Numa topologia do tipo grade retangular exemplifique quais são as vizinhanças de von Neumann e de Moore.
- 2) Na simulação de autômatos celulares 2D o que é o problema de valor de contorno? Como podemos evitá-lo? Explique
- 3) O jogo da vida de Conway é um dos autômatos celulares mais conhecidos e complexos já criados. Quais são as 4 regras de definem a evolução do sistema? Descreva brevemente cada uma.
- 4) Forneça um pseudo-código ou algoritmo que, dado uma célula do autômato  $(i,j)$  na geração atual (considere que  $c[i,j]$  denota a célula em questão), aplique as regras do jogo da vida, atualizando tal célula (considere que  $nc[i,j]$  a célula na nova geração).
- 5) Dadas as configurações iniciais a seguir, evolua o sistema por 5 gerações aplicando as regras do jogo da vida, mostrando as configurações obtidas. O que acontece em cada caso?



- 6) Implemente em Python um simulador para o jogo da vida utilizando para isso um tabuleiro de tamanho mínimo 100 x 100 e um número mínimo de gerações de 200. Teste alguns padrões iniciais e construa animações para visualizar o a evolução do sistema. Utilize o pacote `matplotlib.animation`
- 7) Explique o que fazem os autômatos celulares 2D com as seguintes regras:
  - a) B123/S456
  - b) B3/S23
  - c) B1357/S2468
- 8) Pesquise na internet sobre o modelo de autômato celular conhecido como Brian's brain. O que ele faz?
- 9) Pesquise na internet sobre o modelo de autômato celular conhecido como Langdon's Ant. Descreva sua regra de formação.
- 10) Pesquise sobre o software *Golly - A Game of Life simulator*. Instale o programa e simule diversos modelos de autômatos celulares, dentre os quais:
  - a) Other-Rules → `golly-ants.rle`
  - b) Life-Like → Morley → `enterprise-gun.rle`
  - c) Life → Breeders → `switch-engine-breeder-MR.rle`