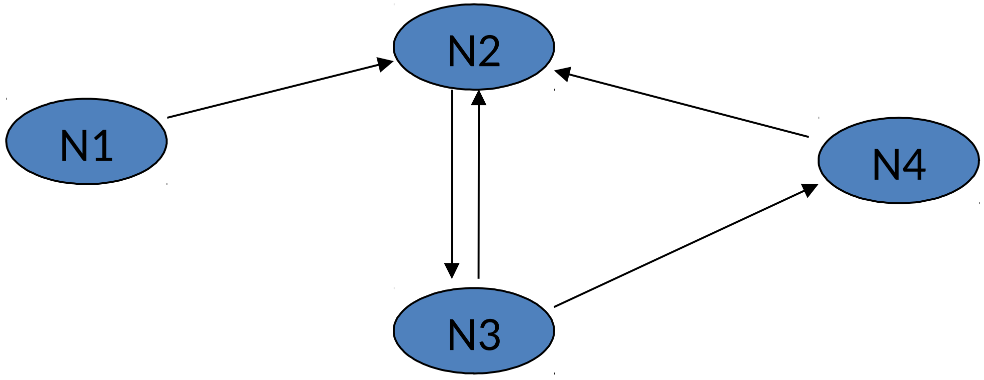
Grafos

Conceitos, representações, algoritmos

Definição de grafo

Um **grafo G=(V,E)** é uma estrutura matemática composta por:

- Um conjunto **V** de vértices ou **nós**
- Um conjunto **E** de pares de nós, designados por ramos, conexões, **ligações** ou **arcos**, cada um ligando dois nós. Estes arcos podem ter uma orientação (pares ordenados) ou não.



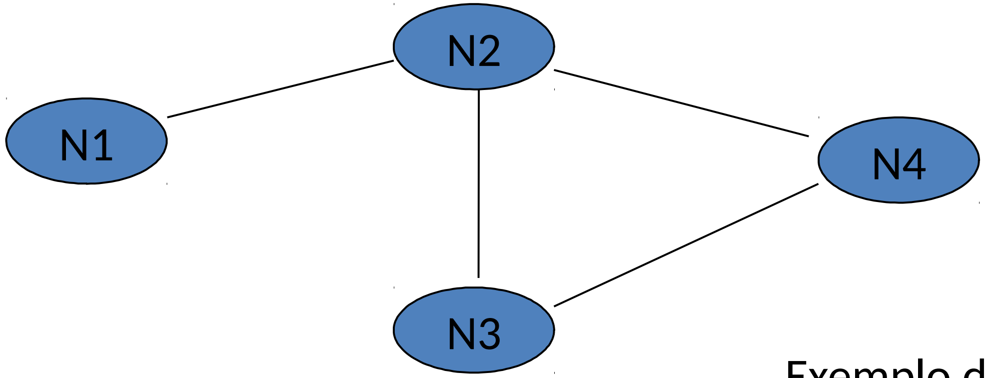
Exemplo de um grafo orientado

V = { N1, N2, N3, N4 }

E = { (N1,N2), (N2, N3), (N3,N2), (N4,N2), (N3,N4) }

Grafos orientados e não orientados

- Grafos **orientados**:
  - Grafos onde as ligações entre os nós (arcos) têm um sentido (elementos de E são pares ordenados)
- Grafos **não orientados**:
  - Grafos onde os arcos não têm um sentido definido (elementos de E são pares não ordenados)



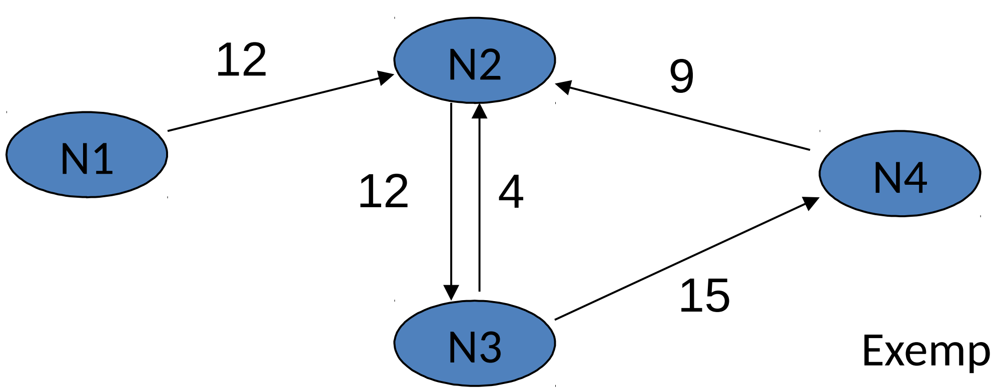
V = { N1, N2, N3, N4 }

E = { (N1,N2), (N2,N3), (N2,N4), (N3,N4) }

Exemplo de um grafo não orientado

Grafos pesados

- Como estruturas de dados, é comum os grafos terem associados aos nós e/ ou aos arcos informação de vários tipos
- No caso mais comum, os arcos podem ter associados valores numéricos, comummente chamados de pesos (grafos **pesados**), sendo os elementos de E tripletos incluindo como último elemento o peso



V = { N1, N2, N3, N4 }

E = { (N1,N2,12), (N3,N2,4), (N2,N3,12), (N4,N2,9), (N3,N4,15) }

Exemplo de um grafo pesado

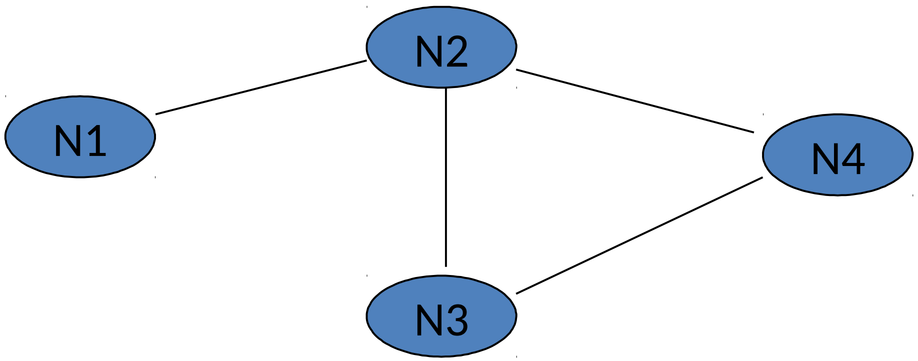
Representações computacionais de grafos

- Os grafos podem ser representados computacionalmente de várias formas distintas
- A escolha depende do tipo de aplicação, do tipo de grafos, da densidade de ligações e dos algoritmos que se pretendem implementar sobre estes
- Representações mais comuns:
  - Matrizes
  - Listas de adjacência

Matrizes de adjacência

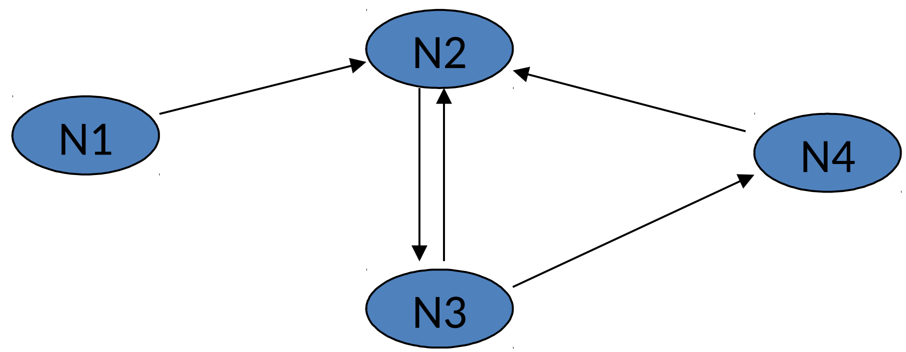
- As matrizes de adjacência representam grafos como **matrizes** em que
  - as linhas correspondem aos nós origem das ligações
  - as colunas correspondem aos nós destino
  - se existe ligação entre nó i e j, então o valor na matriz (M[i][j]) será 1; caso contrário, será 0
  - em grafos não orientados, a matriz é simétrica podendo apenas representar-se uma matriz triangular
  - no caso dos grafos pesados, os pesos nas ligações podem ser representados como valores na matriz
- Esta representação traz vantagens em casos em que a densidade de ligação do grafo é grande, i.e. há um grande nº médio de ligações por nó e quando o nº de nós é baixo
- Em casos onde a densidade é baixa e o nº de nós elevado, a matriz torna-se pesada em termos de memória

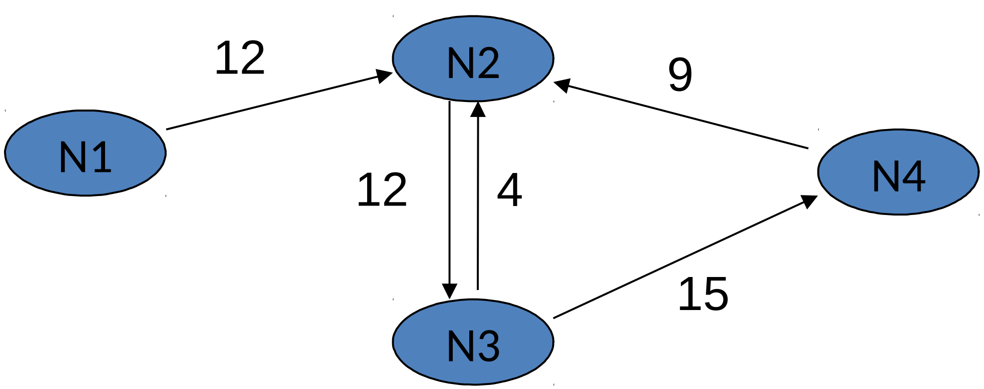
Matrizes de adjacência - exemplos



	N1	N2	N3	N4
N1	0	1	0	0
N2		0	1	1
N3			0	1
N4				0

	N1	N2	N3	N4
N1	0	1	0	0
N2	0	0	1	0
N3	0	1	0	1
N4	0	1	0	0

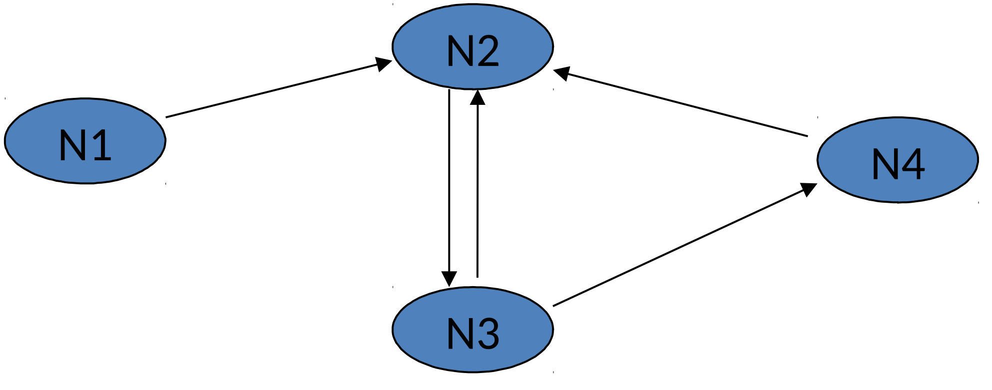




	N1	N2	N3	N4
N1	0	12	0	0
N2	0	0	12	0
N3	0	4	0	15
N4	0	9	0	0

Listas de adjacência

- Apenas se representam as ligações existentes; para cada nó *i* temos uma lista com os arcos originados em *i*; representação mais compacta para grafos com número limitado de ligações.
- Se o grafo é não orientado, poderá haver informação redundante representando-se ambos os sentidos da ligação
- Esta representação pode ser adaptada representando-se cada nó como um objecto e as ligações como listas de referência a objetos desta classe.



Grafo G:

1 => [2]
2 => [3]
3 => [2,4]
4 => [2]

Nós adjacentes, sucessores e antecessores

- Num grafo **orientado** G = (V,E):
  - Um vértice *s* é **sucessor** do vértice *v* se existe em E o par ordenado *(v,s)*
  - Um vértice *p* é **antecessor** do vértice v se existe em E o par ordenado *(p,v)*
  - Se existe em E o par ordenado *(p,s)*, os vértices *p* e *s* dizem-se **adjacentes** (i.e. dois vértices são adjacentes se um é sucessor do outro)
- Num grafo **não orientado** G=(V,E), os vértices x e y são adjacentes se existe em E o par não ordenado (x,y) (i.e. existe (x,y) ou (y,x))

Grau de um nó

- O **grau** de um nó é definido como o número de ligações que ligam esse nó a outros nós, i.e. é o nº de nós adjacentes
- Se grafo é **orientado**, podem definir-se:
  - o grau de **entrada**: número de ligações que chegam a esse nó (nº de predecessores)
  - o grau de **saída**: número de ligações que saem desse nó (nº de sucessores)
- Para o grafo completo pode definir-se:
  - **Grau médio <k>**: média do grau calculada sobre todos os nós
  - **Distribuição do grau P(k)**: probabilidade que um nó tenha grau k, calculadas para cada valor de k = 1, ..., nº de nós do grafo

Caminhos

- Num grafo orientado **G = (V,E)**, um **caminho P** entre dois nós *x* e *y* é definido como uma sequência de nós P_1, P_2, ..., P_n onde $P_1 = x$, $P_n = y$ e todos os pares ordenados (P_i, P_{i+1}) são arcos pertencentes ao grafo (i.e. pertencem a E), sendo **n** o comprimento do caminho P
- Num grafo **não orientado**, a definição é semelhante sendo que a condição é a de que o par não ordenado (P_i, P_{i+1}) pertence ao grafo (note que num par não ordenado não interessa a ordem dos nós)

Nós atingíveis

- Para um dado nó origem v, os nós **atingíveis** são aqueles para os quais existe um caminho com origem em v e final nesse mesmo nó
- Para identificar todos os nós atingíveis de um nó origem é necessário realizar uma **travessia** do grafo, onde o algoritmo passa por:
  - Iniciar no nó origem
  - Visitar todos os sucessores do nó origem, todos os sucessores desses nós, e assim sucessivamente até visitar todos os nós atingíveis

Travessia do grafo

- Duas estratégias podem ser usadas para definir a ordem de exploração dos nós numa travessia:
  - Em **largura**: começa pelo nó origem, depois explora todos os seus sucessores, depois os sucessores destes, e assim sucessivamente até todos os nós atingíveis terem sido explorados
  - Em **profundidade**: começa pelo nó origem e explora o 1º sucessor, seguido pelo 1º sucessor deste e assim sucessivamente até não haver mais sucessores e ter que se fazer “backtracking”

Caminhos mais curtos / distâncias

- O **caminho mais curto** entre dois nós define-se como o caminho P entre esses nós cujo comprimento (nº de nós) seja o menor de entre todos os possíveis caminhos entre eles
- O caminho mais curto pode ser encontrado com uma travessia do grafo em largura terminada quando o nó desejado é atingido
- A **distância** entre dois nós define-se como o comprimento do caminho mais curto entre ambos, i.e. o número de nós visitados (incluindo o destino); se não existir nenhum caminho a distância é considerada infinita

Ciclos e grafos cíclicos/ acíclicos

- Um caminho é chamado de fechado se começa e termina no mesmo nó do grafo
- Um caminho fechado chama-se de **ciclo** (simples) se não tem vértices ou arcos repetidos
- Num grafo orientado, qualquer caminho fechado inclui um ciclo (simples)
- Um grafo que contém ciclos é designado por **cíclico**, enquanto um grafo que não contém ciclos é designado por **acíclico**
- Os grafos orientados acíclicos (designados por DAGs) são bastante importantes e têm propriedades matemáticas únicas (como a ordenação topológica)
- As árvores são casos particulares de grafos acíclicos em que os nós são todos ligados