ANEXO DA AULA 12
Algoritmo do fluxo máximo

A capacidade de cada aresta deve ser um número inteiro positivo.
Input: Uma rede com fonte *a*, destino *z*, capacidade *C*, vértices *a=v0,..,vn=z* e o número de vértices *n*.
Output: Um fluxo máximo *F*

```
procedure max_flow(a,z,C,v,n)
//rótulo de v:(predecessor(v),val(v))
//inicia com fluxo zero
  for cada aresta (i,j) do
    Fij=0
  while true do
    begin
    //remove todos rótulos
    for i:=0 to n do
      begin
      predecessor(vi):=null
      val(vi):=null
      end
    //rotula a
    predecessor(a):= -
    val(a):=∞
     //U é o conjunto dos vértices rotulados não
       examinados
    U:={a}
     // continue até z estar rotulado
     while val(z)=null do
       begin
       if U=ϕ then  // fluxo máx.
          return(F)
       escolha v em U
       U:=U-{v}
       Δ:=val(v)
       for cada aresta (v,w) com val(w)=null do
          if Fvw< Cvw then
            begin
            predecessor(w):=v
            val(w):=min{Δ, Cvw -Fvw }
            U:=U ∪ {w}
            end
       for cada aresta (w,v) com val(w)=null do
          if Fwv>0 then
            begin
            predecessor(w):=v
            val(w):= min{Δ, Fwv }
            U:=U ∪ {w}
            end
       end
    //encontre o caminho P de a a z para
      atualizar o fluxo
    w0:=z
    k:=0
    while wk≠ a do
      begin
         wk+1:=predecessor(wk)
         k:=k+1
      end
    P:=(wk, wk+1,..., w1, w0)
    Δ:=val(z)
    for i=1 to k do
      begin
        e:=(wi, wi-1)
        if e é propriam. orientada em P then
           Fe:= Fe+Δ
        else
           Fe:= Fe- Δ
      end
  end
end max_flow
```