Bellman–Ford (Versão primitiva)

```
for (v = 0; v < G->V; v++)
  { st[v] = -1; wt[v] = maxWT; }
wt[s] = 0; st[s] = 0;
for (i = 0; i < G->V; i++)
  for (v = 0; v < G->V; v++)
    for (t = G->adj[v]; t != NULL; t = t->next)
      if (wt[w = t->v] > wt[v] + t->wt)
        { wt[w] = wt[v] + t->wt; st[w] = v; }
```

Programa 21.9, Bellman–Ford

```
void GRAPHbf(Graph G, int s, int st[], double wt[])
{ int v, w; link t; int N = 0;
  QUEUEinit(G->E);
  for (v = 0; v < G->V; v++) { st[v] = -1; wt[v] = maxWT; }
  wt[s] = 0.0; st[s] = 0;
  QUEUEput(s); QUEUEput(G->V);
  while (!QUEUEempty())
    if ((v = QUEUEget()) == G->V)
      { if (N++ > G->V) return; QUEUEput(G->V); }
    else
      for (t = G->adj[v]; t != NULL; t = t->next)
        if (wt[w = t->v] > wt[v] + t->wt)
          { wt[w] = wt[v] + t->wt;
            QUEUEput(w); st[w] = v; }
}
```