

```

/*****
/* Recherche avec Hill Climbing ( profondeur DFS)
/*****
SOIT
  G UN GRAPHE DE ENTIERS ;
  I UN ENTIER ;
  N , N1 , N2 , N3 , N4 , N5 , N6 DES NOEUDS ( G ) ;
  Pil UNE PILE DE NOEUD ( G ) ;
  Trouv UN BOOLEEN ;
  L UNE LISTE DE ( ENTIER , ENTIER ) ;
  V : ( ENTIER , ENTIER ) ;

DEBUT
{ Un arbre m-aire }
  CREER_GRAPHE ( G , [ N1 ( 5 ) , N2 ( 8 ) , N3 ( 6 ) , N4 ( 3 ) , N5 ( 7 ) , N6 ( 8 ) ]
  UNION [ N1 -> N2 , N1 -> N3 , N1 -> N4 , N3 -> N5 , N3 -> N6 ] ) ;
  CREERPILE ( Pil ) ;
  EMPILER ( Pil , N1 ) ;
  Trouv := FAUX ;
  TQ NON PILEVIDE ( Pil ) ET ( NON Trouv )
    DEPILER ( Pil , N ) ;
    ECRIRE ( 'Noeud : ' , VALEUR ( N ) ) ;
    SI VALEUR ( N ) = 43
      Trouv := VRAI
    SINON
      SI DEGRE ( N ) <> 0
        CREERLISTE ( L ) ;
        POUR I := 1 , DEGRE ( N )
          AFF_STRUCT ( V , 1 , VALEUR ( ADJACENT ( N , I ) ) ) ;
          AFF_STRUCT ( V , 2 , I ) ;
          INSERER ( L , V , I )
        FINPOUR ;
        TRIER_LISTE ( L , 1 ) ;
        POUR I := 1 , DEGRE ( N )
          SI I = 1
            N_IEME ( L , 1 , V ) ;
            EMPILER ( Pil , ADJACENT ( N , STRUCT ( V , 2 ) ) )
          SINON
            SUIVANT ( L , V ) ;
            EMPILER ( Pil , ADJACENT ( N , STRUCT ( V , 2 ) ) )
          FSI
        FINPOUR
      FSI
    FSI
  FSI
  FTQ ;
  SI Trouv
    ECRIRE ( 'S U C C E S' )
  SINON
    ECRIRE ( 'E C H E C' )

```

FSI
FIN