获得的答案 返回

a)

Class- *P* : *P* is class of languages that are decidable in polynomial time on a deterministic single – tape Turing machine. We have to construct an deterministic Turing machine (DTM) to decide *SPATH* in polynomial time.

Let *M* be the *DTM* to decide *SPATH* in polynomial time.

The algorithm of *M* is as follows:

$M =$"on input $\langle G,a,b,k\rangle$ where *m*-node graph *G* has nodes *a* and b:

1. Place a mark "o" on node *a*.

2. for each i from 0 to *m*:

3. If an edge (s,t) is found connecting *s* marked as "*i*" to an unmarked node *t*, mark node *t* with "$i+1$".

4. If *b* is marked with a value at most *k*, accept. Otherwise reject.

This algorithm is similar to PATH algorithm. Here we additionally need to keep the track of length of the shortest paths discovered. That will be done in polynomial time $O(|V|+|E|)$.

Hence, we constructed a *DTM M* to decide *SPATH* in polynomial time.

Therefore, $SPATH+1 \in P$.

(b)

***NP* - complete:** A language *B* is *NP* – complete if it satisfies two conditions.

1. *B* is in *NP* and

2. Every *A* in *NP* is polynomial time reducible to *B*.

To show *LPATH* is *NP* – complete, we need show $LPATH \in NP$ **and** $UHAMPATH \le_p LPATH$

1. $LPATH \in NP$ **:**

We know that "*NP* is the class of languages that have polynomial time verifies.

We construct a verifies *V* for $LPATH$ as follows:

$V =$"on input $\langle G,a,b,k,c\rangle$, where *c* is a path:

1. Check *c* is a non – repeated sequence of nodes in *G*.

2. Check the first term of *c* is a and last is *b*.

3. Check the length of *c* is larger than or equal to *k*.

4. If *c* satisfies the conditions 1 to 3, accept.

5. Otherwise, reject

This verifier *V* can finish in $O(ICI)$ where $|c|$ is the length of *c*.

So, $LPATH \in NP$.

2. $UHAMPATH \le_p LPATH$:

Consider an instant $\langle G,a,b\rangle$ of $UHAMPATH$ problem where $G=\langle V,E\rangle$ is a graph with assigned starting node *a* and ending node *b*.

• The mapping copy $\langle G,a,b\rangle$ and set $k=|V|-1$, then $\langle G,a,b,k\rangle$ is an instance of $LPATH$.

• It can be finished in polynomial time $\left(O\left(|V|+|E|\right)\right)$

• We need to prove $\langle G,a,b\rangle \in UHAMPATH \Leftrightarrow \langle G,a,b,k\rangle \in LPATH$

If $\langle G,a,b\rangle \in UHAMPATH$, then *G* has a Hamiltonian path from *a* to *b*.

• It must be a simple path that goes through every node exactly once,

• Which implies that the length is $|v|-1=k$

• So $\langle G,a,b,k\rangle \in LPATH$

If $\langle G,a,b,k\rangle \in LPATH$, there exists a simple path from *a* and *b* with length $k=|v|-1$.

• Because the graph *G* only has $k+1$ nodes.

• So this simple path must pass through all of needs in graph *G* exactly once.

• So this simple path must be a Hamiltonian path.

• It implies that $\langle G,a,b\rangle \in UHAMPATH$

Therefore, the $LPATH$ is *NP* – complete.