

## Teoría de Complejidad Computacional

Investiga y responde sobre la veracidad (V) o falsedad (F) de los siguientes enunciados relacionados a la Teoría de Complejidad Computacional. Para cada enunciado, justifica tu respuesta (en tus propias palabras). Como evidencia que apoye tu justificación, proveer un enlace de algún artículo, paper y/o post en Internet. Si una pregunta no tiene respuesta válida, indícalo y justifica la razón por la cual o lo consideras no tiene respuesta válida.

Valor: 6 puntos

### NOTAS:

- Algunos de los problemas mencionados son de optimización, pero en clase discutimos como se pueden convertir a problemas de decisión.
- Algunas fuentes de Internet no son confiables. Trata de investigar de múltiples fuentes. Sin embargo, solo debes someter un solo URL.

- 1)   V   2-SUM y 3-SUM son problemas en P y en NP  
Justificación:

2-SUM AND 3- SUM: <https://medium.com/@paulrohan/solving-the-classic-two-sum-and-three-sum-problem-in-javascript-7d5d1d47db03>

2-SUM complejidad  $O(N)$  por lo tanto es P

3-SUM complejidad  $O(N^2 - E)$  donde es una constante por lo tanto es P tambien.

3-SUM es otra variación de subset sum que se puede comprobar en tiempo polinómico tambien y como Son problemas que están en P y se pueden comprobar en tiempo Polinomial,

Sabemos que P Esta dentro de NP.

- 2)   V   N-QUEENS y N-QUEENS-COMPLETION son NP-COMPLETE  
Justificación:

En esta pagina en un post encontré la respuesta:

<https://jair.org/index.php/jair/article/view/11079>

En este archive de Jair se demuestra que N QUEENS y N QUEEN COMPLETION son de NP Complete, se puede ver en el apartado.

- 3)   V   MIN-COIN-CHANGE es NP-Complete y SUBSET-SUM es NP-HARD  
Justificación:

MIN COIN CHANGE: <http://compbio.fmph.uniba.sk/vyuka/eaz/handouts/notes06-np.pdf>

En la página 14 hay una demostración.

SUBSET SUM: <https://en.wikipedia.org/wiki/NP-hardness>, En el primer párrafo segunda línea, está la aclaración.

(AQUÍ TUBE QUE SOMETER DOS URL PARA CADA UNO DE LOS PROBLEMAS)

- 4)   V   GENERALIZED-CHESS es un problema en EXP, pero no en P  
Justificación:

Encontré un libro que no solo habla de este sino de muchos otros problemas, puede ser muy bueno para referencia es este:

[https://books.google.com.do/books?id=CG2MDwAAQBAJ&dq=Generalized+chess+polynomial&hl=es&source=gbs\\_navlinks\\_s](https://books.google.com.do/books?id=CG2MDwAAQBAJ&dq=Generalized+chess+polynomial&hl=es&source=gbs_navlinks_s) en la página 54, párrafo 2 en adelante.

Menciona que solo por unos pocos problemas ha sido posible demostrar que ningún algoritmo puede resolverlo en tiempo polinómico y uno de esos es el Generalized Chess.

El resultado es que no está en P y sí en EXP.

- 5)   V   FRACTIONAL-KNAPSACK es un problema en P, pero 0-1-KNAPSACK no está en P  
Justificación:

FRACTIONAL: <https://cs.nyu.edu/courses/fall02/V22.0310-002/lectures/lecture-24.html>

La complejidad es  $N \log N$  lo que la hace polinomial;

0-1 KNAPSACK no es polinomial debido a que es Pseudopolinomial depende del tamaño de la capacidad.  $O(NW)$ .

LINK: [https://en.wikipedia.org/wiki/Pseudo-polynomial\\_time](https://en.wikipedia.org/wiki/Pseudo-polynomial_time)

(AQUÍ TUBE QUE SOMETER DOS URL PARA CADA UNO DE LOS PROBLEMAS) debido a que son diferentes problemas.

- 6)   V   Existe un problema que es NP-Hard y no NP-Complete  
Justificación:

La respuesta es sencilla son los Halting Problem que no están en NP-Complete

En Wikipedia está la solución: <https://en.wikipedia.org/wiki/NP-hardness>

- 7) \_\_\_F\_\_\_ Existe un problema que es NP-Complete, pero no NP-Hard  
Justificación:

La página de geekforgeeks muy buena por hecho tiene esta solución:

<https://www.geeksforgeeks.org/np-completeness-set-1/>

Es debido a que NP-COMPLETE es subconjunto de NP-HARD.

- 8) \_\_\_NO TIENE Respuesta\_\_\_ Existe un problema en EXP, pero no en NP  
Justificación:

En esta pagina : <https://cs.stackexchange.com/questions/19176/are-there-any-problems-in-complexity-class-exp-that-are-not-in-np>

Es debido a que EXP y NP no se han demostrado que sean diferente.

- 9) \_\_\_NO TIENE RESPUESTA\_\_\_ Existe un problema en NP, pero no es NP-Complete  
Justificación:

Link Misma URL que la siguiente:

[https://upload.wikimedia.org/wikipedia/commons/a/a0/P\\_np\\_np-complete\\_np-hard.svg](https://upload.wikimedia.org/wikipedia/commons/a/a0/P_np_np-complete_np-hard.svg)

No se ha demostrado que NP y NP-Complete sean diferente o iguales mientras no se solucione esto no se puede resolver.

- 10) \_\_\_No tiene respuesta\_\_\_ Existe un problema que es NP-Complete, pero no en P  
Justificación:

Si lo buscas en Wikipedia y ves la siguiente imagen:

[https://upload.wikimedia.org/wikipedia/commons/a/a0/P\\_np\\_np-complete\\_np-hard.svg](https://upload.wikimedia.org/wikipedia/commons/a/a0/P_np_np-complete_np-hard.svg)

Podemos observar que en la imagen que  $P = NP$  y luego menciona que  $P=NP=NP$  complete son iguales se demuestra ahí mismo entonces ya por deducción podemos ver que no se ha podido demostrar esta relación así que no se puede resolver hasta que no se demuestre.

- 11) \_\_\_V\_\_\_ La cantidad de problemas en EXP es mayor que la cantidad en P  
Justificación:

URL: <https://en.wikipedia.org/wiki/EXPTIME>

Es muy simple la justificación P es subconjunto de EXP (: por lo tanto, EXP tiene más v;

12)   F   Todos los problemas EXP-HARD son Undecidable  
Justificación:

LINK: <http://courses.csail.mit.edu/6.892/spring19/lectures/C01.pdf> (Pagina 5)

Sabemos que NP-HARD contiene a EXP-HARD, y tenemos que los problemas Undecidable están en NP-HARD y EXP-HARD extremo no se pueden descartar de ese conjunto, además tenemos el de chess que esta al inicio de los Exp-Hard y al final de los EXP (EXP-COMPLETE) y sabemos que tiene respuesta por lo tanto es falso no todos los problemas de EXP-HARD son Undecidable.