

**SI169 ALGORITMOS AVANZADOS**

**Examen Parcial**

**Ciclo 2014-01**

**Profesores : Marco Bruggmann**

Código de cuadernillo:\_\_\_\_\_\_\_\_\_\_\_\_\_

**Secciones : SW51**

**Duración : 170 Minutos**

No escriba en estos recuadros:

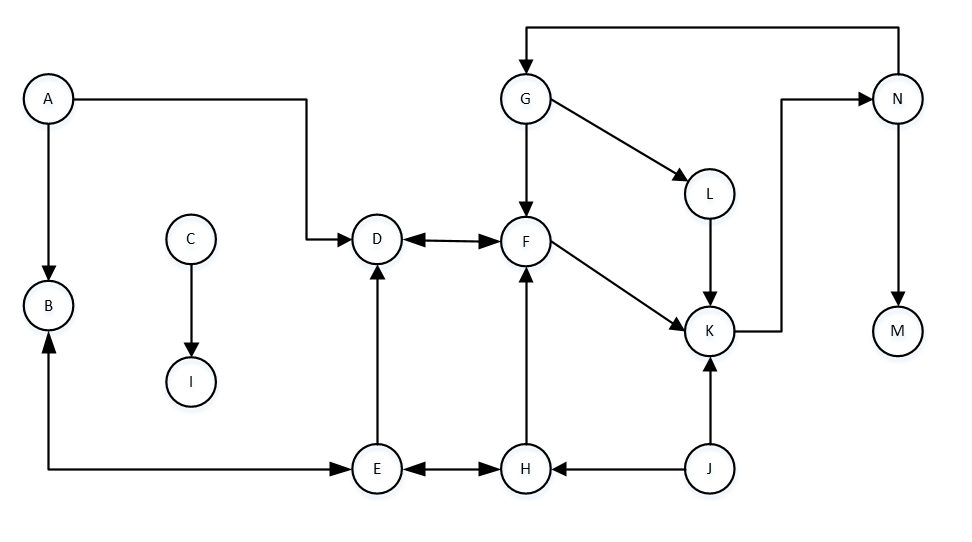
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **1.**  **5,0** | **2.**  **5,0** | **3.**  **5,0** | **4.**  **5.0** | **=** | **20.0** |

##### Indicaciones:

* Lea detenidamente cada pregunta antes de responderla.
* **No** se podrán utilizar libros, apuntes de clase o cualquier otro material impreso.
* Está prohibido el uso de USBs y de Internet. Igualmente, no se permite el préstamo de material alguno.
* ***Coloque el número de su PC en el recuadro superior derecho de esta página***
* ***Al finalizar el examen deberá entregar el texto del examen y copiar sus archivos en la carpeta indicada por el supervisor.***

1. **Grafos**

Teniendo el siguiente grafo *G*, se le pide lo siguiente:

****

* 1. Represente el grafo *G* como una lista de adyacencia. (0.5 ptos)
  2. Halle y grafique los SCC. (2.5 ptos)
  3. Halle el orden topológico de los SCC. (1 pto)
  4. Para cada SCC hallado, ejecute el algoritmo BFS sobre el menor nodo (A<B, B<C, etc) y haga lo siguiente: (1 pto, todo o nada)
     1. Represente gráficamente la progresión.
     2. Muestre en el subgrafo el camino más corto desde el nodo de inicio hacia el resto de nodos del SCC.
     3. Muestre cómo queda la estructura de datos utilizada para cada caso.

1. **Custom-Heapify**

En esta ocasión se pide implementar una función similar a Max-Heapify, pero siguiendo las siguientes reglas:

Si x < 0, entonces el valor de prioridad será x^2-2x-5

Si x > 5, entonces el valor de prioridad será 4x-x^2+10

Sino, el valor de prioridad será x+3.

De su clase de algoritmos avanzados, Ud. recuerda que la función de Heapify es aquella asegura que el Heap cumpla con la propiedad de montículo. Escriba en su cuadernillo el código de la función

void custom\_heapify(A,i,long){

}

Además, usted cuenta con un arreglo *B* en el cual ha almacenado un árbol binario (no necesariamente siguiendo las reglas descritas). Se le pide escribir en su cuadernillo el código que permita convertir el arreglo en un Custom-Heap:

void build\_custom\_heapify(B,long){

}

1. **Publicidad en redes sociales**

Una fuente de ingresos importante de toda red social se obtiene por el cobro de publicidad que se anuncia en la página. Las empresas por lo general tienen dos opciones en este contexto: pagar por click y pagar por impresiones. Una publicación, cuando se trata de pago por click, es cobrada cuando un usuario hace click intencionalmente en el anuncio. En cuanto al pago de impresiones, es cobrado cada vez que al usuario se le muestre la publicación. En este caso, nos centraremos en el cobro por impresión.

Como queremos que la publicidad sea efectiva, tenemos que seleccionar un público objetivo que sea más probable de consumir la publicidad. Para esto, definimos que este conjunto *C* serán los amigos, amigos de amigos, amigos de amigos de amigos y todos aquellos que de alguna u otra forma, tengan relación con un usuario específico (puede haber X grados de separación entre un amigo y otro, pero aun así debe considerarse que existe una relación entre ambos).

El pago por cada impresión es un valor *P,* variable, que es fijado por la empresa (mientras más alto sea el pago, más probable será de aparecer en la página). Otro factor importante son las veces *I* que el usuario navega por la red social, ya que esto equivale al número de impresiones que serán cobradas.

Como experto en algoritmos avanzados, Ud. debe resolver este problema y determinar cuánto tendría que pagar la empresa, si decide lanzar un anuncio sobre un conjunto de amigos específico.

**Entrada:**

Cada caso empieza con tres números:

*N:* que representa el número de registrados en la red

*P*: que es el pago por cada impresión (no necesariamente es un entero)

*R*: que es el número de relaciones amicales en la red social.

Luego, siguen *N* líneas, cada una indicando el valor *I* de cada usuario registrado (el primer valor corresponderá al número de impresiones del primer usuario y así sucesivamente).

Luego, siguen *R* líneas, cada una indicando que el usuario A es amigo del usuario B (y que B es amigo de A).

**Salida:**

Para cada conjunto *C* identificado, imprima el coste total de lanzar la campaña con ese conjunto como público objetivo y el número de personas que visualizarán la publicación.

**Ejemplo de entrada:**

8 0.25 5

10

20

25

5

15

13

18

10

1 2

6 7

8 4

5 8

6 8

**Ejemplo de salida:**

2 7.5

5 15.25

1 6.25

1. **Noche de tragos**

Un amigo suyo acaba de dar su último examen de parciales y decide salir a celebrar con toda la promoción. Como su amigo es responsable y al día siguiente tiene que trabajar por la mañana, Ud. le explica una estrategia para que al día siguiente amanezca sin dolor de cabeza.

La estrategia que le propone es la siguiente: empezar con la bebida de menos contenido alcohólico y luego incrementar progresivamente hasta la bebida con mayor grado. Ud. le explica a su amigo que es muy importante que luego de consumir una bebida B, no puede volver tomar cualquier otra que tenga menor grado alcohólico.

**Entada:**

Cada caso de prueba empieza con un número *N* de tipos de bebidas disponibles y seguido a esto *N* líneas indicando el nombre de cada bebida (de a lo más 51 caracteres cada una). Luego, un número *M* (entre 0 y 200, inclusive) seguido de *M* líneas de la forma B1 B2, indicando que B2 tiene un grado alcohólico mayor a B1 (en este caso, su amigo debería empezar por B1 y luego continuar con B2). Tome en cuenta que estas relaciones son transitivas, es decir que de existir otra relación B2 B3, entonces debería tomar B1 antes de B3.

Cada caso de prueba está separado por una línea en blanco y los casos terminan con un “end of file” EOF.

**Salida:**

Para cada caso, debe imprimir: **Caso #C: Debe tomar las bebidas en este orden: B1 B2 ... BN.**, en donde **C** es el número del caso, empezando en 1 y **B1 B2 ... BN** es la lista de bebidas de tal manera que el contenido alcohólico de **Bi+1** no es menor que el contenido alcohólico de **Bi**. Luego de cada caso de prueba, imprima una línea en blanco.

**Ejemplo de entrada:**

3

vodka

wine

beer

2

wine vodka

beer wine

5

wine

beer

rum

apple-juice

cachaca

6

beer cachaca

apple-juice beer

apple-juice rum

beer rum

beer wine

wine cachaca

10

cachaca

rum

apple-juice

tequila

whiskey

wine

vodka

beer

martini

gin

11

beer whiskey

apple-juice gin

rum cachaca

vodka tequila

apple-juice martini

rum gin

wine whiskey

apple-juice beer

beer rum

wine vodka

beer tequila

**Ejemplo de salida:**

Caso #1: Debe tomar las bebidas en este orden: beer wine vodka.

Caso #2: Debe tomar las bebidas en este orden: apple-juice beer wine rum cachaca.

Caso #3: Debe tomar las bebidas en este orden: apple-juice wine vodka beer rum cachaca tequila whiskey martini gin.