
Práctica de PRO2 - Primavera 2020 - entrega final prorrogadaX76633_es

Este problema es el único canal disponible para realizar **la entrega de la práctica dentro del plazo de la prórroga**. Tened en cuenta que

- El plazo de entrega acaba el domingo 24 de mayo, a las 22h.
- Esta es la entrega que dará lugar a la mayor parte de la nota de la práctica (90%); la nota de la entrega tendrá una parte procedente de corrección automática y otra procedente de corrección manual (consultad el pdf sobre evaluación de la práctica en la web de PRO2)
- Toda entrega de este problema sustituirá a todas las entregas del ejercicio *Práctica de PRO2 - Primavera 2020 - entrega final definitiva*
- Toda entrega de este problema conllevará una penalización equivalente a un juego de pruebas fallido. Por lo tanto, el máximo número computable de juegos de prueba aceptados de cara a la nota automática de la práctica es 3.
- No superar ningún juego de pruebas conllevará un cero en la nota de la entrega
- En esta entrega pedimos más cosas, que detallamos a continuación **en negrita**, además del código de la práctica y el Makefile, de las que se pedían en la entrega provisional
- Pueden realizarse tantas entregas como se quiera, pero solo se tendrá en cuenta la más reciente
- El problema de la entrega provisional seguirá disponible hasta el día anterior al fin del plazo de entrega. De hecho, recomendamos usarlo para probar el código mientras no tengáis preparado el material adicional que pedimos

Observación

El Jutge prueba vuestras entregas mediante 4 juegos de pruebas:

- `sample`: el juego de pruebas público
- `privat1`: combinación de los juegos de pruebas de la entrega intermedia con una situación nueva de eficiencia (gestión de las distancias)
- `privat2`: un poco de todo sobre clusters
- `privat3`: eficiencia de clusters
- `privat3`: eficiencia de kmeros

En un fichero llamado `practica.tar` tenéis que entregar

- Los ficheros `.hh` y `.cc` de las clases y el programa principal **con los comentarios Doxygen correspondientes**. No incluyáis ficheros `.o` y similares.
- El fichero `Makefile`, que usaremos para generar y probar el ejecutable

- Un fichero `html.zip`, obtenido comprimiendo la carpeta `html` del Doxygen generado a partir de los `.hh` y `.cc` anteriores. Dicha documentación no solo ha de incluir la parte pública de las clases (como en la entrega de la especificación) sino también la parte privada (atributos y métodos privados) y la implementación de todas las operaciones

Tened en cuenta las siguientes restricciones:

- El fichero que contiene el programa principal se ha de llamar `program.cc`
- El Makefile ha de generar un ejecutable llamado `program.exe`
- Es importante que uséis las opciones de compilación del Judge de PRO2 (ved Documentation → Compilers → PRO2 a www.jutge.org)
- No usar la opción `-D_GLIBCXX_DEBUG` o usarla de forma incorrecta podrá ser penalizado
- Si no se entrega la documentación generada por el Doxygen, o ésta es incompleta (por ejemplo, si no se pueden ver los elementos privados o la implementación de todas las operaciones), la nota de la corrección manual será cero. Comprobad que vuestra documentación es correcta antes de la entrega. Es necesario usar un Doxyfile basado en el de la sesión 10 de laboratorio para generar correctamente la documentación

Producid el fichero `practica.tar` con la instrucción Linux

```
tar -cvf practica.tar fitxer1 fitxer2 fitxer3 ...
```

desde el directorio/carpeta donde tengáis los ficheros que vais a entregar. Includid esta instrucción en vuestro Makefile, de forma que el `.tar` se pueda generar ejecutando `make practica.tar`. Con eso reduciréis el riesgo de error en sucesivas entregas. El Judge no acepta `.tar` donde los ficheros estén dentro de carpetas. Recomendamos usar GNU `tar` para reducir el riesgo de que el fichero `practica.tar` sea incompatible con el Judge. No es necesario incluir `BinTree.hh` en `practica.tar`.

Ejemplo de entrada

<pre>3 lee_cjt_especies 5 d GCTCCTGTCCGTTTCAGCCG e TACACCATTAACCGGTGGGG a AAAAGATGACCAGCGTAATG b GCAACCTTTGTGGGCGCAGT c ACGATTTGCGTAAGCTATGT imprime_cjt_especies tabla_distancias imprime_arbol_filogenetico crea_especie f GAAAAAAAAAAAAAAAAAAAAA crea_especie g AAAAAAAAAAAAAAAAAAAAAA imprime_cjt_especies tabla_distancias</pre>	<pre>imprime_arbol_filogenetico elimina_especie nuev1 crea_especie nuev1 TTAAACCCGGG inicializa_clusters elimina_especie a imprime_cjt_especies tabla_distancias ejecuta_paso_wpgma imprime_cluster c ejecuta_paso_wpgma imprime_cluster ac</pre>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

```

imprime_cluster eg
imprime_cluster enuevo1
ejecuta_paso_wpgma
ejecuta_paso_wpgma
imprime_cluster ac
crea_especie nuevo2 AATCTGGCTCTGAATAAGACCCAGTATCAAGCCCTACTA
ejecuta_paso_wpgma
ejecuta_paso_wpgma
imprime_cluster ac
inicializa_clusters
crea_especie nuevo2 AAA
imprime_arbol_filogenetico

lee_cjt_especies
0
imprime_cjt_especies
tabla_distancias
imprime_arbol_filogenetico
ejecuta_paso_wpgma
crea_especie pangolin AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
imprime_cjt_especies
tabla_distancias
obtener_gen pangolin
imprime_arbol_filogenetico
ejecuta_paso_wpgma
fin

```

Ejemplo de salida

```

# lee_cjt_especies

# imprime_cjt_especies
a AAAAGATGACCAGCGTAATG
b GCAACCTTTGTGGGCGCAGT
c ACGATTTGCGTAAAGCTATGT
d GCTCCTGTCCGTTTCAGCCG
e TACACCATTAACCGGTGGGG

# tabla_distancias
a: b (90.9091) c (71.4286) d (90.9091) e (90.9091)
b: c (87.5) d (87.5) e (83.871)
c: d (83.871) e (94.1176)
d: e (97.1429)
e:

# imprime_arbol_filogenetico
[(acdbe, 45.7951) [(acd, 43.695) [(ac, 35.7143) [a][c]]]]

# crea_especie f GAAAAAAAAAAAAAAAAAAAAAAAAA
# crea_especie g AAAAAAAAAAAAAAAAAAAAAAAAAA

# imprime_cjt_especies
a AAAAGATGACCAGCGTAATG
b GCAACCTTTGTGGGCGCAGT
c ACGATTTGCGTAAAGCTATGT
d GCTCCTGTCCGTTTCAGCCG
e TACACCATTAACCGGTGGGG
f GAAAAAAAAAAAAAAAAAAAAAAAAA
g AAAAAAAAAAAAAAAAAAAAAAAAAA

# tabla_distancias
a: b (90.9091) c (71.4286) d (90.9091) e (90.9091) f (90.9091) g (90.9091)
b: c (87.5) d (87.5) e (83.871) f (100) g (100)
c: d (83.871) e (94.1176) f (100) g (100)
d: e (97.1429) f (100) g (100)
e: f (100) g (100)
f: g (9.52381)
g:

# imprime_arbol_filogenetico
[(acdbefg, 49.6528) [(acdbe, 45.7951) [(acd, 43.695) [(a)[c]]]]]]

# elimina_especie nuevo1
ERROR: La especie nuevo1 no existe.

# crea_especie nuevo1 TTAAACCCGGG

# inicializa_clusters
a: b (90.9091) c (71.4286) d (90.9091) e (90.9091) f (90.9091) g (90.9091)
b: c (87.5) d (87.5) e (83.871) f (100) g (100) nuevo1 (92.3077)
c: d (83.871) e (94.1176) f (100) g (100) nuevo1 (92.3077)
d: e (97.1429) f (100) g (100) nuevo1 (92.3077)
e: f (100) g (100) nuevo1 (66.6667)
f: g (9.52381) nuevo1 (96.5517)
g: nuevo1 (96.5517)
nuevo1:

```

```

# elimina_especie a

# imprime_cjt_especies
b GCAACCTTTGTGGGCGCAGT
c ACGATTTCGTAAGCTATGT
d GCTCCTGTCCGTTTCAGCCG
e TACACCATTAACCGTG GGG
f GAAAAAAAAAAAAAAAAAAAAA
g AAAAAAAAAAAAAAAAAAAAAA
nuevo1 TTAAACCCGGG

# tabla_distancias
b: c (87.5) d (87.5) e (83.871) f (100) g (100)
c: d (83.871) e (94.1176) f (100) g (100)
d: e (97.1429) f (100) g (100) nuevo1 (92.3077)
e: f (100) g (100) nuevo1 (66.6667)
f: g (9.52381) nuevo1 (96.5517)
g: nuevo1 (96.5517)
nuevo1:

# ejecuta_paso_wpgma
a: b (90.9091) c (71.4286) d (90.9091) e (90.9091) fg (100)
b: c (87.5) d (87.5) e (83.871) fg (100)
c: d (83.871) e (94.1176) fg (100) nuevo1 (92.3077)
d: e (97.1429) fg (100) nuevo1 (92.3077)
e: fg (100) nuevo1 (66.6667)
fg: nuevo1 (96.5517)
nuevo1:

# imprime_cluster c
[c]

# ejecuta_paso_wpgma
a: b (90.9091) c (71.4286) d (90.9091) enuevo1 (89.4444) fg (100)
b: c (87.5) d (87.5) enuevo1 (83.6022) fg (100)
c: d (83.871) enuevo1 (93.2127) fg (100)
d: enuevo1 (94.7253) fg (100)
enuevo1: fg (98.2759)
fg:

# imprime_cluster ac
ERROR: El cluster ac no existe.

# imprime_cluster eg
ERROR: El cluster eg no existe.

# imprime_cluster enuevo1
[(enuevo1, 33.3333) [e][nuevo1]]

# ejecuta_paso_wpgma
ac: b (89.2045) d (87.39) enuevo1 (91.3333) fg (100)
b: d (87.5) enuevo1 (83.6022) fg (100)
d: enuevo1 (94.7253) fg (100)
enuevo1: fg (98.2759)
fg:

# ejecuta_paso_wpgma
ac: benuevo1 (90.2691) d (87.39) fg (97.2222)
benuevo1: d (91.1126) fg (99.1379)
d: fg (100)
fg:

# imprime_cluster ac
[(ac, 35.7143) [a][c]]

# crea_especie nuevo2 AATCTGGCTCTGAATAAGACCCAGTATCAAGC

# ejecuta_paso_wpgma
acd: benuevo1 (90.6909) fg (98.6111)
benuevo1: fg (99.1379)
fg:

# ejecuta_paso_wpgma
acdbenuevo1: fg (98.8745)
fg (100) nuevo1 (83.3333)
nuevo1 (92.3077)

# imprime_cluster ac
ERROR: El cluster ac no existe.

# inicializa_clusters
b: c (87.5) d (87.5) e (83.871) f (100) g (100) nuevo1 (92.3077)
c: d (83.871) e (94.1176) f (100) g (100) nuevo1 (92.3077)
d: e (97.1429) f (100) g (100) nuevo1 (92.3077) nuevo2 (85.4167)
e: fg (100) fg (100) nuevo1 (96.5517) nuevo2 (98.2143)
fg (92.3077) (96.5517) nuevo2 (100)
nuevo1: nuevo2 (90.6977)
nuevo2:

# crea_especie nuevo2 AAA
ERROR: La especie nuevo2 ya existe.

# imprime_arbol_filogenetico
[(bnuevo2enuevo1cdfg, 49.7287) [(bnuevo2enuevo1cd, 45.1429) [c][fg]
enuevo1 (89.4444) fg (94.4444) fg (100)]

# imprime_cjt_especies

# tabla_distancias

# imprime_arbol_filogenetico
ERROR: El conjunto de clusters es vacio.

# ejecuta_paso_wpgma
ERROR: num_clusters <= 1

# crea_especie pangolin AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

# imprime_cjt_especies
pangolin AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

# tabla_distancias
pangolin:

# obtener_gen pangolin
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

# imprime_arbol_filogenetico
[pangolin]

# ejecuta_paso_wpgma
ERROR: num_clusters <= 1

```

Información del problema

Autor : PRO2

Generación : 2020-05-29 13:35:05

© *Jutge.org*, 2006–2020.

<https://jutge.org>