

Errata *Bioinformatics for Evolutionary Biologists. A Problems Approach*

Bernhard Haubold & Angelika Börsch-Haubold

April 8, 2021

1. p. 4, Problem 8: The line

```
You cannot tune a mouse but you can tuna fish  
should be
```

```
You can't tune a mouse but you can tuna fish
```

2. p. 5, Table 1.1: “Delete character left of cursor” → Delete character right of cursor
3. p. 8, Problem 24: The single line

```
cp ~/Downloads/data.tgz . tar -xvzf data.tgz
```

Should be two lines

```
cp ~/Downloads/data.tgz .  
tar -xvzf data.tgz
```

4. p. 15, Problem 53: “echo 5” → seq 5
 5. p. 17, Problem 66: “User filter.sed” → Use filter.sed
 6. p. 18, Problem 68: “press C-c C-c” → press C-c once or, on some systems, twice
 7. p. 20, Problem 79: “END {” → END{
 8. p. 36, Problem 132: “The usage of this program is explained in the header of the script.” → Its help function is invoked
- ```
awk -f dotPlotFilter.awk -v h=1
```
9. p. 37, Problem 139: “(2012, 0), (2012, 150)” → (2021, 0), (2021, 150)
  10. p. 63, Problem 221: “Bwt” → Bwt
  11. p. 76, Problem 263: -subjectdmAdhAdhdup.fasta → subject dmAdhAdhdup.fasta
  12. p. 86, “New Programs, Help” samtools → samtools
  13. p. 87, Problem 325: “-reference” → --reference
  14. p. 92, Problem 344: “sed' s/MG\_/g'” → sed 's/MG\_/g'
  15. p. 96, Problem 357: “use human  $\beta$ -hemoglobin as the subject” → use human  $\beta$ -hemoglobin as the query
  16. p. 99, Problem 365: “200000” → 20000

17. p. 107, Problem 389: Add “Look up their names in `hominidae.fasta`.”
18. p. 109, New Program: “Comment” → Source
19. p. 115, Problem 416: The assignment
- ```
seed = $RANDOM
```
- should not contain blanks
- ```
seed=$RANDOM
```
20. p. 117, Problem 420: “unique” → `uniq`
21. p. 120, Problem 427: “100by” → `100 by`
22. p. 121, Chapter Intro: “from the preset” → from the present
23. p. 122, Problem 430: See 19.
24. p. 123, Problem 433: See 19.
25. p. 123, Problem 433: “leftmost leaf” → rightmost leaf
26. p. 124, Problem 433: “leftmost leaf” → rightmost leaf
27. p. 124, Algorithm 3: As written, it only works for 0-based arrays. However, we are using 1-based arrays; hence it should be
- Require:**  $n$  {sample size}
- Require:** `tree` {array of  $n$  leaves followed by  $n - 1$  internal nodes}
- Ensure:** Tree topology
- 1: **for**  $i \leftarrow n$  **to** 2 **do**
  - 2:    $p \leftarrow i \times \text{ran}() + 1$  { $1 \leq p \leq i$ }
  - 3:   `tree[ $2 \times n - i + 1$ ].child1`  $\leftarrow$  `tree[ $p$ ]`
  - 4:   `tree[ $p$ ]`  $\leftarrow$  `tree[ $i$ ]` {Replace child by the rightmost entry in the “leafy” part of tree}
  - 5:    $p \leftarrow (i - 1) \times \text{ran}() + 1$  { $1 \leq p \leq i - 1$ }
  - 6:   `tree[ $2 \times n - i + 1$ ].child2`  $\leftarrow$  `tree[ $p$ ]`
  - 7:   `tree[ $p$ ]`  $\leftarrow$  `tree[ $2 \times n - i + 1$ ]` {Replace child by parent}
  - 8: **end for**
28. p. 124f, Problem 436: Add “Use it to compute the expected number of segregating sites for  $n = 4$  and  $\theta = 10$ .”
29. p. 140, Answer 8: “`C-c` twice.” → `C-c`.
30. p. 150, Answer 61: “`mgGenes3.txt`” → `mgGenes.txt`
31. p. 151, Answer 67: Add “To actually count the gyrases, add `| wc -l`.”
32. p. 152, Answer 71: “`$4-/[+]/`” → `$4~/[+]/`
33. p. 156, Answer 84: The last alignment should be
- ```
ACCGT-
AT--GT
```
34. p. 157, Answer 88: The second alignment should be
- ```
Query: 1 ACCGT 5
 | ||
Sbjct: 1 AT-GT 4
```

35. p. 158, Answer 94: “whereorf.gp” → where orf.gp

36. p. 160, Answer 102: Change

```
cut -f 2 |
histogram |
```

to

```
cut -f 2 |
head -n 10000 |
histogram |
```

37. p. 161, Answer 107: “\*0.05” → /20

38. p. 162, Answer 109: “Visualize the result” → Where NR is the number of lines (records) read. Visualize the result

39. p. 171, Answer 125: The AWK program should be

```
awk '{c++;if($6>0)print c, $6}'
```

40. p. 175f, Answer 139: “Next” → Alternatively, we could generate a single column of numbers and then arrange them in pairs by replacing

```
awk -f break.awk > cdsDm.txt
```

with

```
sed 's/ /\n/g' |
paste - -
```

Next

41. p. 187, Answer 172: In naive.awk “i<=n-m” should be i<=n-m+1 and the indentation was garbled; is should be

```
BEGIN{
 n = split(t, ta, "")
 m = split(p, pa, "")
 for(i=1; i<=n-m+1; i++){
 for(j=1; j<=m; j++){
 if(ta[i+j-1] != pa[j])
 break
 }
 if(j == m + 1)
 print i
 }
}
```

42. p. 187, Answer 173: The line

```
if(j == m + 1)
```

can be expressed more succinctly as

```
if(j > m)
```

43. p. 193, Answer 187: “ $\{\emptyset\}$ ”  $\rightarrow \emptyset$
44. p. 222, Answer 283: “where 41 is the last overlapping position”  $\rightarrow$  where 35 is the last overlapping position
45. p. 246, bottom of page: delete spurious “pagebreak”
46. p. 258, Answer 384: The lines

```
f = 1
for (j=3; j<=(2*i-3); j+=2)
 f *= j
```

can be expressed more succinctly

```
f *= 2 * i - 3
```

47. p. 272, Answer 418: Extend commands by “gv wrapWf.ps &” to display the figure.
48. p. 283, Answer 458:

- “2 5 10 20 50 100 200 500”  $\rightarrow$  2 5 10 20 50
- “-m 7 -d 2.5”  $\rightarrow$  -m 12 -d 3.5