Perl for Biologists

Session 1
March 4, 2015

Introduction

Jaroslaw Pillardy
Organization

• Perl for Biologists consists of 15 sessions, one every week, until June 10th

• Sessions will be taught by different Bioinformatics Facility staff members, the speakers are listed on the workshop web pages

• Slides will be posted online before each session.

• Please feel free to contact us with any questions:
  o Workshop coordinator: Jaroslaw Pillardy jp86@cornell.edu, Rhodes 623
  o Each session’s speaker name is listed on session web page
  o You can find us in the Bioinformatics Facility directory [http://cbsu.tc.cornell.edu/staff.aspx](http://cbsu.tc.cornell.edu/staff.aspx)


• No programming experience necessary.
Organization

- BioHPC Lab machines are reserved for you and available all the time between now (March 4th) and June 21st (end of day June 20th)

- Please DO NOT use them for extensive calculations. It is fine to run on them any “light” Perl-related calculations, create and test Perl programs etc.

- You can see your reservations after logging into BioHPC Lab website [http://cbsu.tc.cornell.edu/](http://cbsu.tc.cornell.edu/)

- Helpful links:
  - Lab Users guide [http://cbsu.tc.cornell.edu/lab/use.aspx](http://cbsu.tc.cornell.edu/lab/use.aspx)
  - My reservations [http://cbsu.tc.cornell.edu/lab/labresman.aspx](http://cbsu.tc.cornell.edu/lab/labresman.aspx)
  - Reset password [http://cbsu.tc.cornell.edu/lab/labpassreset.aspx](http://cbsu.tc.cornell.edu/lab/labpassreset.aspx)

- Useful books:
  - “Learning Perl”, Randal Schwartz, Brain D Foy, Tom Phoenix
  - “Beginning Perl for Bioinformatics”, James Tisdall
Organization

“Perl for Biologists” office hours will be held each Tuesday 11am-1pm and 3pm-4pm in 623 Rhodes.

Please don’t hesitate to come if you have any questions or want to further discuss course topics.
Organization

• The workshop has practical examples and exercises.

• You can follow examples during the lecture, or you can carry them out afterwards.

• If you have any problems with them contact us or come to office hours

• **The only way to learn programming is to try!** Please do after lecture exercises – they are always discussed at the beginning of the next session.

• You can practice Perl programming on any computer, including your Windows or Mac laptop.

• We will focus on our Linux machines since it is most likely environment on which you will run your future Perl programs.

• Therefore next few slides are “Linux primer”.
Connecting to Linux b machines

Text-based connection: ssh (Secure SHell)

GUI (graphical) connection: X-Windows or VNC

Please refer to the following document for more information about GUI connections:
http://cbsu.tc.cornell.edu/lab/doc/Introduction_to_BioHPC_Lab_v2.pdf
Logging in to a Linux machine

- On any Linux machine, you need
  - **network name** of the machine (e.g. cbsumm10.tc.cornell.edu)
  - an **account**, i.e., **user ID** and **password**
  - on your local computer: remote access software
    (typically: **ssh client**)

- Linux is a multiple-access system: multiple users may be logged in and operate on one machine at the same time
Logging in to a Linux machine

- Remotely from a PC via ssh client

  - Install and configure remote access software (PuTTY).
  
  - Use PuTTY to open a terminal window on the reserved workstation using ssh protocol;
  
  - You may open several terminal windows, if needed.
Logging in to a Linux machine

- Remotely from other Linux machine or Mac via native ssh client

  - Launch the Mac’s terminal window. Type

    ssh jarekp@cbsuwrkstX.tc.cornell.edu

    (replace the “cbsuwrkstX” with the workstation that you just reserved, and “jarekp” with your own user ID). Enter the lab password when prompted.

  - You may open several terminal windows, if needed, and log in to the workstation from each of them.
Logging in to CBSU machines from outside of Cornell

Two ways to connect from outside:

- Install and run the CIT-recommended the VPN software [http://www.it.cornell.edu/services/vpn](http://www.it.cornell.edu/services/vpn) to join the Cornell network, then proceed as usual.

- Log in to cbsulogin.tc.cornell.edu (or cbsulogin2.tc.cornell.edu):
  
  ```
  ssh jarekp@cbsulogin.tc.cornell.edu  
  ```
  
  (using PuTTy or other ssh client program)

Once logged in to cbsulogin, ssh further to your reserved machine

```
ssh jarekp@cbsuwrkst3.tc.cornell.edu
```

Backup login machine is cbsulogin2.tc.cornell.edu

https://cbsu.tc.cornell.edu/lab/doc/BioHPCLabexternal.pdf
Terminal window

- User communicates with the machine via **commands** typed in the terminal window

  - Commands are interpreted by a program referred to as **shell** – an interface between Linux and the user. We will be using the shell called **bash** (another popular shell is **tcsh**).

  - Typically, each command is typed in one line and “entered” by hitting the **Enter** key on the keyboard.

  - Commands deal with **files** and **processes**, e.g.,
    - request information (e.g., list user’s files)
    - launch a simple task (e.g., rename a file)
    - start an application (e.g., Firefox web browser, BWA aligner, IGV viewer, …)
    - stop an application
Logging out of a Linux machine

While in terminal window, type exit or Ctrl-D - this will close the current terminal window
How to access BioHPC Lab machines

BioHPC Lab User’s Guide
http://cbsu.tc.cornell.edu/lab/userguide.aspx

Slides from workshop “Introduction to BioHPC Lab”
http://cbsu.tc.cornell.edu/lab/doc/Introduction_to_BioHPC_Lab_v2.pdf

Slides from workshop “Linux for Biologists”
http://cbsu.tc.cornell.edu/lab/doc/Linux_workshop_Part2.pdf
Programming languages

- **Strongly typed** vs. **Loosely typed (context based)**
  - All variables declared
  - Variables interpreted dynamically
  - C, C++, Java, C#
  - Perl, Python, Visual Basic

- **Scripted (interpreted)** vs. **Compiled**
  - Executed “on the fly”, by line
  - Binary version of code executed
  - Perl, Visual Basic, Shell
  - Python, Java, C#
  - C, C++, Fortran

- **Flat** vs. **Object oriented**
  - No complex objects
  - Objects with properties and functions
  - C, Pascal
  - Perl, Java, C#, C++
Programming languages

Perl is a loosely typed, interpreted, object-oriented programming language.

Loosely typed:
Easier to write, more flexible, no need for extra code to “cast” variables. VERY EASY to make errors. Perl variables are typed dynamically based on context.

Interpreted:
More portable – will execute anywhere where interpreter is present IF program does not require specific libraries and IF it doesn’t use system specific commands. MUCH slower, automatic code optimization impossible.

Object-oriented:
Program can be compartmentalized with reusable code. Very powerful way to solve problems. Slower.
Why Perl?

• Easy to learn, fast to write (rapid prototyping), informal

• High-level – compact code, lots of useful functions

• Huge public library of code available that can be directly used

• Runs anywhere (with some caution)

• Flexible: useful for scripting, websites as well as large programs

• Perl is not fast, but excellent to “stich” together other programs – very good for pipelines, task automation, interacting with OS.

• Perl can be easily used to perform various “in-between” functions like process control, file/data control and conversion, string operations, database operations and many more
Programming cycle

- EDIT / DESIGN
- VERIFY / COMPILE
- RUN / TEST
Perl programs are scripts – text files interpreted line by line

Need to use TEXT editor to create and edit them

TEXT file is a file than uses only letters, numbers and common symbols plus “new line” or “tab” special characters. NO formatting or other binary code (MS Word vs. text example).

Plain ASCII characters: byte codes between 32 and 126 (byte => 8 bits, 0-255; 1 bit => smallest unit of information)

Modern text files can use special characters (e.g. ó or ö) and symbols (e.g. β or §) with Unicode – and Perl can work with them too. But they MUST be used with a TEXT editor (and better yet – not used at all 😊)

Example: Notepad and Word
### ASCII Table

<table>
<thead>
<tr>
<th>Dec</th>
<th>Hx</th>
<th>Oct</th>
<th>Char</th>
<th>Dec</th>
<th>Hx</th>
<th>Oct</th>
<th>Html</th>
<th>Chr</th>
<th>Dec</th>
<th>Hx</th>
<th>Oct</th>
<th>Html</th>
<th>Chr</th>
<th>Dec</th>
<th>Hx</th>
<th>Oct</th>
<th>Html</th>
<th>Chr</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>000</td>
<td>NUL</td>
<td>32</td>
<td>20</td>
<td>040</td>
<td>#32;</td>
<td>Space</td>
<td>64</td>
<td>40</td>
<td>100</td>
<td>#64;</td>
<td>@</td>
<td>96</td>
<td>60</td>
<td>140</td>
<td>#96;</td>
<td>`</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>001</td>
<td>SOH</td>
<td>33</td>
<td>21</td>
<td>041</td>
<td>#33;</td>
<td>!</td>
<td>65</td>
<td>41</td>
<td>101</td>
<td>#65;</td>
<td>A</td>
<td>97</td>
<td>61</td>
<td>141</td>
<td>#97;</td>
<td>a</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>002</td>
<td>STX</td>
<td>34</td>
<td>22</td>
<td>042</td>
<td>#34;</td>
<td>&quot;</td>
<td>66</td>
<td>42</td>
<td>102</td>
<td>#66;</td>
<td>B</td>
<td>98</td>
<td>62</td>
<td>142</td>
<td>#98;</td>
<td>b</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>003</td>
<td>ETX</td>
<td>35</td>
<td>23</td>
<td>043</td>
<td>#35;</td>
<td>#</td>
<td>67</td>
<td>43</td>
<td>103</td>
<td>#67;</td>
<td>C</td>
<td>99</td>
<td>63</td>
<td>143</td>
<td>#99;</td>
<td>c</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>004</td>
<td>EOT</td>
<td>36</td>
<td>24</td>
<td>044</td>
<td>#36;</td>
<td>$</td>
<td>68</td>
<td>44</td>
<td>104</td>
<td>#68;</td>
<td>D</td>
<td>100</td>
<td>64</td>
<td>144</td>
<td>#100;</td>
<td>d</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>005</td>
<td>ENQ</td>
<td>37</td>
<td>25</td>
<td>045</td>
<td>#37;</td>
<td>%</td>
<td>69</td>
<td>45</td>
<td>105</td>
<td>#69;</td>
<td>E</td>
<td>101</td>
<td>65</td>
<td>145</td>
<td>#101;</td>
<td>e</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>006</td>
<td>ACK</td>
<td>38</td>
<td>26</td>
<td>046</td>
<td>#38;</td>
<td>&amp;</td>
<td>70</td>
<td>46</td>
<td>106</td>
<td>#70;</td>
<td>F</td>
<td>102</td>
<td>66</td>
<td>146</td>
<td>#102;</td>
<td>f</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>007</td>
<td>BEL</td>
<td>39</td>
<td>27</td>
<td>047</td>
<td>#39;</td>
<td>'</td>
<td>71</td>
<td>47</td>
<td>107</td>
<td>#71;</td>
<td>G</td>
<td>103</td>
<td>67</td>
<td>147</td>
<td>#103;</td>
<td>g</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>010</td>
<td>BS</td>
<td>40</td>
<td>28</td>
<td>050</td>
<td>#40;</td>
<td>(</td>
<td>72</td>
<td>48</td>
<td>110</td>
<td>#72;</td>
<td>H</td>
<td>104</td>
<td>68</td>
<td>150</td>
<td>#104;</td>
<td>h</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>011</td>
<td>TAB</td>
<td>41</td>
<td>29</td>
<td>051</td>
<td>#41;</td>
<td>)</td>
<td>73</td>
<td>49</td>
<td>111</td>
<td>#73;</td>
<td>I</td>
<td>105</td>
<td>69</td>
<td>151</td>
<td>#105;</td>
<td>i</td>
</tr>
<tr>
<td>10</td>
<td>A</td>
<td>012</td>
<td>LF</td>
<td>42</td>
<td>2A</td>
<td>052</td>
<td>#42;</td>
<td>*</td>
<td>74</td>
<td>4A</td>
<td>112</td>
<td>#74;</td>
<td>J</td>
<td>106</td>
<td>6A</td>
<td>152</td>
<td>#106;</td>
<td>j</td>
</tr>
<tr>
<td>11</td>
<td>B</td>
<td>013</td>
<td>VT</td>
<td>43</td>
<td>2B</td>
<td>053</td>
<td>#43;</td>
<td>+</td>
<td>75</td>
<td>4B</td>
<td>113</td>
<td>#75;</td>
<td>K</td>
<td>107</td>
<td>6B</td>
<td>153</td>
<td>#107;</td>
<td>k</td>
</tr>
<tr>
<td>12</td>
<td>C</td>
<td>014</td>
<td>FF</td>
<td>44</td>
<td>2C</td>
<td>054</td>
<td>#44;</td>
<td>,</td>
<td>76</td>
<td>4C</td>
<td>114</td>
<td>#76;</td>
<td>L</td>
<td>108</td>
<td>6C</td>
<td>154</td>
<td>#108;</td>
<td>l</td>
</tr>
<tr>
<td>13</td>
<td>D</td>
<td>015</td>
<td>CR</td>
<td>45</td>
<td>2D</td>
<td>055</td>
<td>#45;</td>
<td>-</td>
<td>77</td>
<td>4D</td>
<td>115</td>
<td>#77;</td>
<td>M</td>
<td>109</td>
<td>6D</td>
<td>155</td>
<td>#109;</td>
<td>m</td>
</tr>
<tr>
<td>14</td>
<td>E</td>
<td>016</td>
<td>SO</td>
<td>46</td>
<td>2E</td>
<td>056</td>
<td>#46;</td>
<td>.</td>
<td>78</td>
<td>4E</td>
<td>116</td>
<td>#78;</td>
<td>N</td>
<td>110</td>
<td>6E</td>
<td>156</td>
<td>#110;</td>
<td>n</td>
</tr>
<tr>
<td>15</td>
<td>F</td>
<td>017</td>
<td>SI</td>
<td>47</td>
<td>2F</td>
<td>057</td>
<td>#47;</td>
<td>/</td>
<td>79</td>
<td>4F</td>
<td>117</td>
<td>#79;</td>
<td>O</td>
<td>111</td>
<td>6F</td>
<td>157</td>
<td>#111;</td>
<td>o</td>
</tr>
<tr>
<td>16</td>
<td>10</td>
<td>020</td>
<td>DLE</td>
<td>48</td>
<td>30</td>
<td>060</td>
<td>#48;</td>
<td>0</td>
<td>80</td>
<td>50</td>
<td>120</td>
<td>#80;</td>
<td>P</td>
<td>112</td>
<td>70</td>
<td>160</td>
<td>#112;</td>
<td>p</td>
</tr>
<tr>
<td>17</td>
<td>11</td>
<td>021</td>
<td>DC1</td>
<td>49</td>
<td>31</td>
<td>061</td>
<td>#49;</td>
<td>1</td>
<td>81</td>
<td>51</td>
<td>121</td>
<td>#81;</td>
<td>Q</td>
<td>113</td>
<td>71</td>
<td>161</td>
<td>#113;</td>
<td>q</td>
</tr>
<tr>
<td>18</td>
<td>12</td>
<td>022</td>
<td>DC2</td>
<td>50</td>
<td>32</td>
<td>062</td>
<td>#50;</td>
<td>2</td>
<td>82</td>
<td>52</td>
<td>122</td>
<td>#82;</td>
<td>R</td>
<td>114</td>
<td>72</td>
<td>162</td>
<td>#114;</td>
<td>r</td>
</tr>
<tr>
<td>19</td>
<td>13</td>
<td>023</td>
<td>DC3</td>
<td>51</td>
<td>33</td>
<td>063</td>
<td>#51;</td>
<td>3</td>
<td>83</td>
<td>53</td>
<td>123</td>
<td>#83;</td>
<td>S</td>
<td>115</td>
<td>73</td>
<td>163</td>
<td>#115;</td>
<td>s</td>
</tr>
<tr>
<td>20</td>
<td>14</td>
<td>024</td>
<td>DC4</td>
<td>52</td>
<td>34</td>
<td>064</td>
<td>#52;</td>
<td>4</td>
<td>84</td>
<td>54</td>
<td>124</td>
<td>#84;</td>
<td>T</td>
<td>116</td>
<td>74</td>
<td>164</td>
<td>#116;</td>
<td>t</td>
</tr>
<tr>
<td>21</td>
<td>15</td>
<td>025</td>
<td>NAK</td>
<td>53</td>
<td>35</td>
<td>065</td>
<td>#53;</td>
<td>5</td>
<td>85</td>
<td>55</td>
<td>125</td>
<td>#85;</td>
<td>U</td>
<td>117</td>
<td>75</td>
<td>165</td>
<td>#117;</td>
<td>u</td>
</tr>
<tr>
<td>22</td>
<td>16</td>
<td>026</td>
<td>SYN</td>
<td>54</td>
<td>36</td>
<td>066</td>
<td>#54;</td>
<td>6</td>
<td>86</td>
<td>56</td>
<td>126</td>
<td>#86;</td>
<td>V</td>
<td>118</td>
<td>76</td>
<td>166</td>
<td>#118;</td>
<td>v</td>
</tr>
<tr>
<td>23</td>
<td>17</td>
<td>027</td>
<td>ETB</td>
<td>55</td>
<td>37</td>
<td>067</td>
<td>#55;</td>
<td>7</td>
<td>87</td>
<td>57</td>
<td>127</td>
<td>#87;</td>
<td>W</td>
<td>119</td>
<td>77</td>
<td>167</td>
<td>#119;</td>
<td>w</td>
</tr>
<tr>
<td>24</td>
<td>18</td>
<td>030</td>
<td>CAN</td>
<td>56</td>
<td>38</td>
<td>070</td>
<td>#56;</td>
<td>8</td>
<td>88</td>
<td>58</td>
<td>130</td>
<td>#88;</td>
<td>X</td>
<td>120</td>
<td>78</td>
<td>170</td>
<td>#120;</td>
<td>x</td>
</tr>
<tr>
<td>25</td>
<td>19</td>
<td>031</td>
<td>EM</td>
<td>57</td>
<td>39</td>
<td>071</td>
<td>#57;</td>
<td>9</td>
<td>89</td>
<td>59</td>
<td>131</td>
<td>#89;</td>
<td>Y</td>
<td>121</td>
<td>79</td>
<td>171</td>
<td>#121;</td>
<td>y</td>
</tr>
<tr>
<td>26</td>
<td>1A</td>
<td>032</td>
<td>SUB</td>
<td>58</td>
<td>3A</td>
<td>072</td>
<td>#58;</td>
<td>:</td>
<td>90</td>
<td>5A</td>
<td>132</td>
<td>#90;</td>
<td>Z</td>
<td>122</td>
<td>7A</td>
<td>172</td>
<td>#122;</td>
<td>z</td>
</tr>
<tr>
<td>27</td>
<td>1B</td>
<td>033</td>
<td>ESC</td>
<td>59</td>
<td>3B</td>
<td>073</td>
<td>#59;</td>
<td>;</td>
<td>91</td>
<td>5B</td>
<td>133</td>
<td>#91;</td>
<td>[</td>
<td>123</td>
<td>7B</td>
<td>173</td>
<td>#123;</td>
<td>{</td>
</tr>
<tr>
<td>28</td>
<td>1C</td>
<td>034</td>
<td>FS</td>
<td>60</td>
<td>3C</td>
<td>074</td>
<td>#60;</td>
<td>&lt;</td>
<td>92</td>
<td>5C</td>
<td>134</td>
<td>#92;</td>
<td>\</td>
<td>124</td>
<td>7C</td>
<td>174</td>
<td>#124;</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>1D</td>
<td>035</td>
<td>GS</td>
<td>61</td>
<td>3D</td>
<td>075</td>
<td>#61;</td>
<td>=</td>
<td>93</td>
<td>5D</td>
<td>135</td>
<td>#93;</td>
<td>]</td>
<td>125</td>
<td>7D</td>
<td>175</td>
<td>#125;</td>
<td>}</td>
</tr>
<tr>
<td>30</td>
<td>1E</td>
<td>036</td>
<td>RS</td>
<td>62</td>
<td>3E</td>
<td>076</td>
<td>#62;</td>
<td>&gt;</td>
<td>94</td>
<td>5E</td>
<td>136</td>
<td>#94;</td>
<td>^</td>
<td>126</td>
<td>7E</td>
<td>176</td>
<td>#126;</td>
<td>~</td>
</tr>
<tr>
<td>31</td>
<td>1F</td>
<td>037</td>
<td>US</td>
<td>63</td>
<td>3F</td>
<td>077</td>
<td>#63;</td>
<td>?</td>
<td>95</td>
<td>5F</td>
<td>137</td>
<td>#95;</td>
<td>DEL</td>
<td>127</td>
<td>7F</td>
<td>177</td>
<td>#127;</td>
<td>DEL</td>
</tr>
</tbody>
</table>

Source: www.LookupTables.com
### ASCII Table

<table>
<thead>
<tr>
<th>Decimal</th>
<th>Character</th>
<th>Decimal</th>
<th>Character</th>
<th>Decimal</th>
<th>Character</th>
<th>Decimal</th>
<th>Character</th>
<th>Decimal</th>
<th>Character</th>
<th>Decimal</th>
<th>Character</th>
<th>Decimal</th>
<th>Character</th>
<th>Decimal</th>
<th>Character</th>
<th>Decimal</th>
<th>Character</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>Ć</td>
<td>144</td>
<td>É</td>
<td>160</td>
<td>á</td>
<td>176</td>
<td></td>
<td>192</td>
<td>L</td>
<td>208</td>
<td></td>
<td>224</td>
<td>α</td>
<td>240</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>129</td>
<td>ũ</td>
<td>145</td>
<td>æ</td>
<td>161</td>
<td>í</td>
<td>177</td>
<td></td>
<td>193</td>
<td>ñ</td>
<td>209</td>
<td></td>
<td>225</td>
<td>β</td>
<td>241</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>130</td>
<td>é</td>
<td>146</td>
<td>Æ</td>
<td>162</td>
<td>ó</td>
<td>178</td>
<td></td>
<td>194</td>
<td>Ñ</td>
<td>210</td>
<td></td>
<td>226</td>
<td>Γ</td>
<td>242</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>131</td>
<td>â</td>
<td>147</td>
<td>Õ</td>
<td>163</td>
<td>ú</td>
<td>179</td>
<td></td>
<td>195</td>
<td>Õ</td>
<td>211</td>
<td></td>
<td>227</td>
<td>π</td>
<td>243</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>132</td>
<td>ä</td>
<td>148</td>
<td>ò</td>
<td>164</td>
<td>ň</td>
<td>180</td>
<td></td>
<td>196</td>
<td>Ń</td>
<td>212</td>
<td></td>
<td>228</td>
<td>Σ</td>
<td>244</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>133</td>
<td>à</td>
<td>149</td>
<td>Ø</td>
<td>165</td>
<td>Ñ</td>
<td>181</td>
<td></td>
<td>197</td>
<td>Ñ</td>
<td>213</td>
<td></td>
<td>229</td>
<td>σ</td>
<td>245</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>134</td>
<td>å</td>
<td>150</td>
<td>Õ</td>
<td>166</td>
<td>à</td>
<td>182</td>
<td></td>
<td>198</td>
<td>À</td>
<td>214</td>
<td></td>
<td>230</td>
<td>µ</td>
<td>246</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>135</td>
<td>ç</td>
<td>151</td>
<td>Ù</td>
<td>167</td>
<td>ç</td>
<td>183</td>
<td></td>
<td>199</td>
<td>Ù</td>
<td>215</td>
<td></td>
<td>231</td>
<td>τ</td>
<td>247</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>136</td>
<td>è</td>
<td>152</td>
<td>Ý</td>
<td>168</td>
<td>ë</td>
<td>184</td>
<td></td>
<td>200</td>
<td>Ý</td>
<td>216</td>
<td></td>
<td>232</td>
<td>Φ</td>
<td>248</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>137</td>
<td>ø</td>
<td>153</td>
<td>Ø</td>
<td>169</td>
<td>Õ</td>
<td>185</td>
<td></td>
<td>201</td>
<td>Ø</td>
<td>217</td>
<td></td>
<td>233</td>
<td>⊙</td>
<td>249</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>138</td>
<td>å</td>
<td>154</td>
<td>Ü</td>
<td>170</td>
<td>å</td>
<td>186</td>
<td></td>
<td>202</td>
<td>Ü</td>
<td>218</td>
<td></td>
<td>234</td>
<td>Ω</td>
<td>250</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>139</td>
<td>í</td>
<td>155</td>
<td>Ï</td>
<td>171</td>
<td>í</td>
<td>187</td>
<td></td>
<td>203</td>
<td>Ï</td>
<td>219</td>
<td></td>
<td>235</td>
<td>δ</td>
<td>251</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>140</td>
<td>î</td>
<td>156</td>
<td>Ë</td>
<td>172</td>
<td>å</td>
<td>188</td>
<td></td>
<td>204</td>
<td>Ë</td>
<td>220</td>
<td></td>
<td>236</td>
<td>∞</td>
<td>252</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>141</td>
<td>ì</td>
<td>157</td>
<td>Ì</td>
<td>173</td>
<td>ì</td>
<td>189</td>
<td></td>
<td>205</td>
<td>Ì</td>
<td>221</td>
<td></td>
<td>237</td>
<td>ϕ</td>
<td>253</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>142</td>
<td>Ä</td>
<td>158</td>
<td>È</td>
<td>174</td>
<td>Å</td>
<td>190</td>
<td></td>
<td>206</td>
<td>Å</td>
<td>222</td>
<td></td>
<td>238</td>
<td>ε</td>
<td>254</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>143</td>
<td>Å</td>
<td>159</td>
<td>Œ</td>
<td>175</td>
<td>»</td>
<td>191</td>
<td></td>
<td>207</td>
<td>»</td>
<td>223</td>
<td></td>
<td>239</td>
<td>⊙</td>
<td>255</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Source:** [www.LookupTables.com](http://www.LookupTables.com)
TEXT Editors

**vi**
- Available on all UNIX-like systems (Linux included), i.e., also on lab workstations (type `vi` or `vi file_name`)
- Free Windows implementation available (once you learn vi, you can just use one editor everywhere)
- Runs locally on Linux machine (no network transfers)
- User interface rather peculiar (no nice buttons to click, need to remember quite a few keyboard commands instead)
- Some love it, some hate it

**gvim**
- Vi (see above) with a graphical interface – X-Windows needed. Windows version available.

**nano**
- Available on most Linux machines (our workstations included; type `nano` or `nano file_name`)
- Intuitive user interface. Keyboard commands-driven, but help always displayed on bottom bar (unlike in vi).
- Runs locally on Linux machine (no network transfers during editing)
TEXT Editors

**gedit** (installed on lab workstations; just type *gedit* or *gedit file_name* to invoke)
- X-windows application – need to have X-ming running on client PC.
- **May be slow on slow networks**...

**edit+** ([http://www.editplus.com/](http://www.editplus.com/))
- Commercial product
- Runs on a local machine (laptop) and transfers data to/from Linux workstation as needed
- Can browse Linux directories in a Windows-like file explorer
- **May be slow on slow networks**
- Some people swear by it

**emacs** (installed on lab workstations)

**Xcode** (Mac)

**Notepad** (Windows)
TEXT Files on Unix, Windows and Mac

End-of-line problem:

• Unix: \n CR 10 0x0a
• Windows \n\r CR+LF 10 13 0x0a 0x0d
• Mac (old) \r LF 13 0x0d
• Mac (new) \n CR 10 0x0a

Make sure files transferred from one system to another are properly converted

On Linux there is a set of nice utilities

unix2dos file_name

dos2unix file_name

unix2mac file_name

mac2unix file_name

Example: Windows and Unix files on Windows
Vi basics

Opening a file:

vi my_reads.fastq (open the file my_reads.fastq in the current directory for editing; if the file does not exist, it will be created)

Command mode: typing will issue commands to the editor (rather than change text itself)
Edit mode: typing will enter/change text in the document

<Esc> exit edit mode and enter command mode (this is the most important key – use it whenever you are lost)

The following commands will take you to edit mode:

i enter insert mode
r single replace
R multiple replace
a move one character right and enter insert mode
o start a new line under current line
O start a new line above the current line

The following commands operate in command mode (hit <Esc> before using them)

x delete one character at cursor position
dd delete the current line
G go to end of file
1G go to beginning of file
154G go to line 154
$ go to end of line
1 go to beginning of line
:q! exit without saving
:w save (but not exit)
:wq! save and exit

Arrow keys: move cursor around (in both modes)
Look of a typical Perl script:

```perl
#!/usr/local/bin/perl

#this is my first Perl script

print "Hello, CBSU\n";
```
```
#!/usr/local/bin/perl

#this is my first Perl script

print "Hello, CBSU\n";
```

“shebang” notation – path to the program to interpret the script, must be the first line and start with #!

anything starting with # is a comment, unless it is #! in the first line

statement ends with a semicolon

function to print out text
"shebang" notation – path to the program to interpret the script, must be the first line and start with #!

```
#!/usr/local/bin/perl

#this is my first Perl script

print("Hello, CBSU\n");
```

anything starting with # is a comment, unless it is #! in the first line

parentheses can be always omitted, unless it changes the meaning of expression

statement ends with semicolon

function to print out text
Strings in Perl

- Sequence of characters – simple (ASCII) or extended (Unicode, wide)

- Special characters like NL or CR are represented as \xxxx (C notation)
  - \n  new line (NL)
  - \t  tab character
  - \r  return (CR)
  - \x0a  any character represented by hex number (0a = 10 = NL)
  - \"  double quotation
  - \'  single quotation
  - \\
  backslash

- Strings may be joined by ‘.’ operator

  "string 1 " . "string 2"  <=>  "string 1 string 2"

- Some characters have special meaning in Perl, most prominently  $ and @
  - \$  {dollar}
  - \@  {at}
Strings in Perl

• Single Quoted

Single quoted strings have LITERAL meaning – no special characters are recognized:

'string 1'  string 1
'string 1\n'  string 1{backslash}n
'\string 1\'  'string 1'
' string 1\\1 '  string 1\1

• Double-Quoted

Double quoted strings do interpret special characters properly:

"string 1\n"  string 1{new line}
"\"string 1\\""  "string 1"
Perl installation and usage depends on the OS

External Perl libraries (modules) are accessible via CPAN

CPAN = Comprehensive Perl Archive Network

You can download and use any of publicly available modules in your programs
Perl on Linux

• Almost always installed as a part of the system, if not ask your system admin

• Usually it is /usr/bin/perl or /usr/local/bin/perl

• May be several versions installed, each with its own libraries and features

• Version can be checked with command
  >perl -v
  >/usr/bin/perl -v

• If you need a particular Perl installation in your program, write it into the first line
  #!/usr/local/special/bin/perl

• If you need default Perl installation in your program, write it into the first line
  #!/usr/bin/env perl

• Once invoked, Perl interpreter knows where its system-wide modules reside
Perl on Linux

Execute Perl program

- If the scripts has executable right
  
  >./script_name.pl
  
  >./script_name.pl >& output

- Regardless of executable right
  
  >perl script_name

- Compile (verify) Perl program
  
  >perl -c script_name

Make script executable:

>chmod u+x script_name
Perl on Linux

If you need custom modules located in a custom place:

• write it into first line

```bash
#!/usr/local/bin/perl -I /home/jarekp/my_modules
```

• set environmental variable

```bash
PERL5LIB=/home/jarekp/my_modules:/usr/another/path/lib; export PERL5LIB
```

• Execute explicitly with Perl interpreter and options

```bash
>perl -I /home/jarekp/my_modules my_script.pl
```
Lets write and execute the script NOW

```perl
#!/usr/local/bin/perl

#this is my first Perl script

print "Hello, CBSU\n";
```
Perl on Linux: CPAN

Two interfaces to CPAN

> cpan

> perl -MCPAN -e shell

Then you can type command

install modname - install module modname
r modname - report if upgrade is available
upgrade modname - upgrade
m modname - info about modname

Remember: there is a cpan for EACH Perl installation, make sure you are using the right one
Perl on Linux: CPAN

If you want to install a module for your own use, without being an admin:

Configure cpan (only first time)

> cpan
  o conf makepl_arg INSTALL_BASE=~/.myPERL_LIB
  o conf mbuild_arg INSTALL_BASE=~/.myPERL_LIB
  o conf prefs_dir ~/.myPERL_LIB/prefs
  o conf commit

Install module(s)

> cpan
  install modname

Set up environment so Perl knows where to look

PERL5LIB=/home/jarekp/myPERL_LIB/lib/perl5:$PERL5LIB
Export PERL5LIB

Need to reset CPAN:

  o conf init
Perl on Linux: CPAN

Local configuration example

Configure cpan (only first time)

```
> cpan
  o conf makepl_arg INSTALL_BASE=/home/jarekp/perl5
  o conf mbuild_arg INSTALL_BASE=/home/jarekp/perl5
  o conf prefs_dir /home/jarekp/perl5/prefs
  o conf commit
```

Set up environment so Perl knows where to look: edit /home/jarekp/.bashrc and add the following

```
export PERL_LOCAL_LIB_ROOT="$PERL_LOCAL_LIB_ROOT:/home/jarekp/perl5";
export PERL_MB_OPT="--install_base /home/jarekp/perl5";
export PERL_MM_OPT="INSTALL_BASE=/home/jarekp/perl5";
export PERL5LIB="/home/jarekp/perl5/lib/perl5:$PERL5LIB";
export PATH="/home/jarekp/perl5/bin:$PATH";
```
Perl on Windows

Recommended Perl is ActivePerl: http://www.activestate.com/activeperl

Download binary and install – choose free version.

“shebang” line of any script is ignored on Windows

Windows recognizes Perl scripts by extension .pl

There is a nice GUI to CPAN

Example of script and GUI
Perl on Mac

Similarly as on Linux it comes preinstalled on OS X.

All Linux information should apply.
A bit more complicated script

#!/usr/local/bin/perl

use warnings;
use Bio::Perl;

#this is my first Perl script

print "Hello, CBSU\n";
use ModuleName;
Declares usage of Perl module “ModuleName”, includes all proper definitions

use warnings;
Declares use of “warnings” module – Perl will now report any place it thinks is ambiguous or suspicious: same as >perl –w

use Bio::Perl;
Declares use of BioPerl module – more details later
“use” statement can be declared as a parameter of Perl interpreter

>perl -MBio::Perl

... and then something can be executed ...

>perl -MBio::Perl -e "print "OK\n";"

If Bio::Perl is installed it will print "OK", otherwise an error will occur.

Easy way to check if a module is installed.

Example: CPAN installation of Template::HTML
Exercises

1. Write a Perl program that prints your name and e-mail in the following format in one line:
   first_name last_name <emailaddr@domain.edu>

2. Are the following modules installed on your BioHPC Lab machine?
   - Net::Ping
   - XML::Special
   - Net::Telnet
   - CBSU::HDF5