# Lab1

*Vergil*

*October 8, 2015*

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com (http://rmarkdown.rstudio.com).

Installing R & RStudio

```
system("who",intern=T)
```

```
## [1] "minjiezhang console  Oct 14 07:56 "
```

Installing Packages and Loading them using library()

```
find.package("MASS")
```

```
## [1] "/Library/Frameworks/R.framework/Versions/3.2/Resources/library/MASS"
```

```
find.package("XML")
```

```
## [1] "/Library/Frameworks/R.framework/Versions/3.2/Resources/library/XML"
```

```
find.package("foreign")
```

```
## [1] "/Library/Frameworks/R.framework/Versions/3.2/Resources/library/foreign"
```

2. Application of Basic R Rules

```
27^8
```

```
## [1] 282429536481
```

```
50119-7891
```

```
## [1] 42228
```

```
4^(2/3)
```

```
## [1] 2.519842
```

```
32^(2/5)
```

```
## [1] 4
```

```
22614+98849+49371+1734+11043+74432
```

```
## [1] 258043
```

2.2 Vectors 2.2.1 Numeric Vectors

A.

```
junk <- c(1,2,4,5,3,5,3,5,3,5)
junk
```

```
##  [1] 1 2 4 5 3 5 3 5 3 5
```

B.

```
junk2 <- seq(from=1,to=999,by=2)
junk2
```

```
##   [1]   1   3   5   7   9  11  13  15  17  19  21  23  25  27  29  31  33
##  [18]  35  37  39  41  43  45  47  49  51  53  55  57  59  61  63  65  67
##  [35]  69  71  73  75  77  79  81  83  85  87  89  91  93  95  97  99 101
##  [52] 103 105 107 109 111 113 115 117 119 121 123 125 127 129 131 133 135
##  [69] 137 139 141 143 145 147 149 151 153 155 157 159 161 163 165 167 169
##  [86] 171 173 175 177 179 181 183 185 187 189 191 193 195 197 199 201 203
## [103] 205 207 209 211 213 215 217 219 221 223 225 227 229 231 233 235 237
## [120] 239 241 243 245 247 249 251 253 255 257 259 261 263 265 267 269 271
## [137] 273 275 277 279 281 283 285 287 289 291 293 295 297 299 301 303 305
## [154] 307 309 311 313 315 317 319 321 323 325 327 329 331 333 335 337 339
## [171] 341 343 345 347 349 351 353 355 357 359 361 363 365 367 369 371 373
## [188] 375 377 379 381 383 385 387 389 391 393 395 397 399 401 403 405 407
## [205] 409 411 413 415 417 419 421 423 425 427 429 431 433 435 437 439 441
## [222] 443 445 447 449 451 453 455 457 459 461 463 465 467 469 471 473 475
## [239] 477 479 481 483 485 487 489 491 493 495 497 499 501 503 505 507 509
## [256] 511 513 515 517 519 521 523 525 527 529 531 533 535 537 539 541 543
## [273] 545 547 549 551 553 555 557 559 561 563 565 567 569 571 573 575 577
## [290] 579 581 583 585 587 589 591 593 595 597 599 601 603 605 607 609 611
## [307] 613 615 617 619 621 623 625 627 629 631 633 635 637 639 641 643 645
## [324] 647 649 651 653 655 657 659 661 663 665 667 669 671 673 675 677 679
## [341] 681 683 685 687 689 691 693 695 697 699 701 703 705 707 709 711 713
## [358] 715 717 719 721 723 725 727 729 731 733 735 737 739 741 743 745 747
## [375] 749 751 753 755 757 759 761 763 765 767 769 771 773 775 777 779 781
## [392] 783 785 787 789 791 793 795 797 799 801 803 805 807 809 811 813 815
## [409] 817 819 821 823 825 827 829 831 833 835 837 839 841 843 845 847 849
## [426] 851 853 855 857 859 861 863 865 867 869 871 873 875 877 879 881 883
## [443] 885 887 889 891 893 895 897 899 901 903 905 907 909 911 913 915 917
## [460] 919 921 923 925 927 929 931 933 935 937 939 941 943 945 947 949 951
## [477] 953 955 957 959 961 963 965 967 969 971 973 975 977 979 981 983 985
## [494] 987 989 991 993 995 997 999
```

C.

```
a <- function(m) {(c(m:0,1:m))^2}
```

D.

```
junk^2/100
```

```
##  [1] 0.01 0.04 0.16 0.25 0.09 0.25 0.09 0.25 0.09 0.25
```

E.

```
b <- function(x) {c(x[1:3],log(x[4]),x[5],log(x[6]),x[7:10])};b(junk)
```

```
##  [1] 1.000000 2.000000 4.000000 1.609438 3.000000 1.609438 3.000000
##  [8] 5.000000 3.000000 5.000000
```

2.2.2 Character(String) and Logical Vectors (A)

```
junk3 <- c("wrath","avarice","sloth","pride","lust","envy","gluttony")
junk3
```

```
## [1] "wrath"    "avarice"  "sloth"    "pride"    "lust"     "envy"
## [7] "gluttony"
```

B.

```
junk3[c(4,7)]
```

```
## [1] "pride"    "gluttony"
```

C. In the first case, because of the ordering character>numeric>logical, when an operation involves logical values and numeric values, R converts the logical values to numeric values with TRUE=1 and FALSE=0. Therefore, the result in the first case is converted to (1,1,0)+3=(4,4,3)

In the second case, however,because the matrix operation involves character values, which are greater value type than logical values, the logical values in b are converted to character values, namely “TRUE”, “TRUE”, “FALSE”. Therefore, when we try to add 3 to these character values, it no longer works

2.3 Matrices (A)

```
matrix(seq(from=26,to=50,by=1),nrow=5,byrow = T)
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]   26   27   28   29   30
## [2,]   31   32   33   34   35
## [3,]   36   37   38   39   40
## [4,]   41   42   43   44   45
## [5,]   46   47   48   49   50
```

```
matrix(seq(from=26,to=50,by=1),ncol=5,byrow = T)
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]   26   27   28   29   30
## [2,]   31   32   33   34   35
## [3,]   36   37   38   39   40
## [4,]   41   42   43   44   45
## [5,]   46   47   48   49   50
```

```
t(matrix(seq(from=26,to=50,by=1),ncol=5))
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]   26   27   28   29   30
## [2,]   31   32   33   34   35
## [3,]   36   37   38   39   40
## [4,]   41   42   43   44   45
## [5,]   46   47   48   49   50
```

B.

```
matrix(seq(from=26,to=50,by=1),nrow=5,byrow = T)[3,]
```

```
## [1] 36 37 38 39 40
```

C.

```
matrix(seq(from=26,to=50,by=1),nrow=5,byrow = T)[,c(2,5)]
```

```
##      [,1] [,2]
## [1,]   27   30
## [2,]   32   35
## [3,]   37   40
## [4,]   42   45
## [5,]   47   50
```

2.4 Data Frames (A)

```
data(BOD)
```

B.

```
class(BOD)
```

```
## [1] "data.frame"
```

```
mode(BOD)
```

```
## [1] "list"
```

C.

```
str(BOD)
```

```
## 'data.frame':    6 obs. of  2 variables:
##  $ Time  : num  1 2 3 4 5 7
##  $ demand: num  8.3 10.3 19 16 15.6 19.8
##  - attr(*, "reference")= chr "A1.4, p. 270"
```