An Algorithmic Approach to the Detection and Prevention of Plagiarism

Karl J. Ottenstein

Report Number:
76-200
AN ALGORITHMIC APPROACH TO THE
DETECTION AND PREVENTION OF
PLAGIARISM

Karl J. Ottenstein
Computer Sciences Department
Purdue University
West Lafayette, Indiana 47907

CSD-TR 200
August 1976
The significant problem of detecting (nearly) identical student homework papers is non-trivial since a grader for a large class cannot remember all previously graded papers while examining the current one. This problem can be reduced by quantifying papers in such a way that equivalent ones are given equal values. Here we discuss one possible quantification which works well when applied to student computer programs.

The desired quantification is a function which maps the "homework space" into some value space. The ideal function, \( f \), would impose a partitioning on the set of papers in the sense that if \( x \) and \( y \) are homework papers and the \( P_i \) are partitions of the homework space with \( x \in P_i \) and \( y \in P_j \) then \( i = j \) iff \( f(x) = f(y) \). If \( f(x) = f(y) \) and \( x \) and \( y \) are unique, then one of \( x \) and \( y \) is a plagiarized version of the other. In other words, when all partitions have but one element, no cheating has occurred. This ideal function is unobtainable for several reasons: it is possible for identical work to be performed independently, the semantic equivalence of two items cannot always be shown deterministically, and there is a subjective area between plagiarism and paraphrasing.

Our task, then, is to find a good approximation to this function. The approximation should at least map all potentially equivalent homework papers into the same partition. It may not guarantee accuracy in that two papers being in the same partition will not imply that they are necessarily plagiarized. If \( P_1, P_2, \ldots, P_{n} \) are the ideal partitions, our approximation should create \( Q_1, Q_2, \ldots, Q_{m} \) where each \( Q_i \) is either some \( P_j \) or the union of several \( P_{j}'s \). That is, the partitions are merely cruder.

The constant functions satisfy our requirements for an approximation since only one partition will be created; but, they do not simplify our initial problem since all elements must be individually inspected for cheating. A function which maps a homework paper into the integer representing its length in characters will invariably create numerous partitions, but they will not be the desired \( Q_i \); the replacement of one token by a
synonym of a different length will place plagiarized assignments in separate partitions. A length function based on the number of tokens would eliminate this problem, but will still group together totally unrelated assignments simply because they have the same length. A function which takes into account some measure of the information content of a homework paper should give us more accurate partitions.

Any meaningful language can have its symbols classified into three sets:

- operators
- operands
- "syntactic sugar": symbols used only for readability

The information content of an element of a language, then, depends on the operators and operands, some function of which should lead to a good approximation to our ideal partitioning. This is simply a more formal description of the approach employed by [Bulut 1973].

In his study of student FORTRAN programs, Bulut counted the basic software science [Halstead 1972, 1977] parameters:

- $i_1$: the number of unique operators
- $i_2$: the number of unique operands
- $N_1$: the total number of occurrences of operators
- $N_2$: the total number of occurrences of operands

He noted that "the probability of using $i_1$ and $i_2$ symbols exactly $N_1$ and $N_2$ times in two different expressions is very slim." Plagiarized copies were found by hand checking programs with identical $i_1$, $i_2$, $N_1$, and $N_2$ values. Bulut observed that, as with the length function above, the results of this method are not affected by changes to operand names since such changes will not modify $i_2$ or $N_2$.

A program to count these four parameters for FORTRAN modules was written [Otteinstein 1976] and used to confirm Bulut's work. Table 1 shows the partitioning imposed on 47 student programs from CS 210 at Purdue University by the 'software science method'. In the formalism developed here, we consider this method a mapping of programs into 4-tuples, $(i_1, i_2, N_1, N_2) \in N \times N \times N \times N$, where $N$ denotes the set of natural numbers. Two partitions (A and B) have two programs in them; the rest have one. One program in partition B is a copy of the other, with slightly different comments and margining. The other pair is not as immediately detectable as being plagiarized because one author apparently changed all of the variable names and label numbers. Other programs with close correspondence of the parameters were
compared, but without positive results. Thus it seems that a
good partitioning was obtained. Copies of the programs in
partition A are included in Appendix A with the parameter counts.

The size of a program (in tokens) is given as column N in
Table 1. Since $N=N_1+N_2$, the partitions created by the length
function mentioned above are supersets of those created by the
software science method. Here, the length function creates 10
partitions of size greater than one, while the software science
method seems to have given us the ideal partitioning. So at
least in this case, the additional information provided by $\gamma_1$ and
$\gamma_2$ is well worth the small effort required to obtain it.

Bulut called the chances of two student programs having equal
4-tuples "slim". We can get a more quantitative probability
estimate by observing that $\gamma_1$, $\gamma_2$, $N_1$, and $N_2$ all appear to have
somewhat normal distributions, in agreement with our intuition.
(Appendix B gives the histograms for the four parameters.) In
our particular sample, we have:

<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>median</th>
<th>mode</th>
<th>s.d.</th>
<th>min</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma_1$</td>
<td>17.00</td>
<td>17</td>
<td>15</td>
<td>2.07</td>
<td>13</td>
<td>24</td>
</tr>
<tr>
<td>$\gamma_2$</td>
<td>35.38</td>
<td>35</td>
<td>35</td>
<td>3.93</td>
<td>27</td>
<td>45</td>
</tr>
<tr>
<td>$N_1$</td>
<td>145.77</td>
<td>139</td>
<td>135</td>
<td>19.30</td>
<td>116</td>
<td>195</td>
</tr>
<tr>
<td>$N_2$</td>
<td>111.36</td>
<td>106</td>
<td>101</td>
<td>17.26</td>
<td>84</td>
<td>154</td>
</tr>
</tbody>
</table>

Assuming this normal distribution, there is clearly a greater
likelihood of finding a pair of independently written programs
with equal parameter values near the means as there is of finding
such a pair with values on the tails. Thus, we can be more
confident of a partition's accuracy as its individual parameter
values approach the tails of their distribution curves.

Since the four parameters are not mutually independent, we use
a multivariate normal density function, $g$, determined by the
means vector, $\mu=(17.00, 35.38, 145.77, 111.36)$, and the
covariance matrix $\Sigma$, $g(x) = (x - \mu)^T \Sigma^{-1} (x - \mu)$, to get a feel for the closeness of a 4-tuple to the means
vector. The expression $g(x)/g(\mu)$ is 1 at $x=\mu$ and approaches 0 as
we move away from the mean. Evaluated at the 4-tuple $X$ for
partitions $A$ and $B$, this expression results in 0.45 and 0.018,
respectively. This indicates that the programs in partition $B$
are very probably plagiarized (the partition is accurate), while
those in $A$ are less probably so. Visual inspection of the
programs is clearly warranted in any case, but one would be
particularly suspicious of those in partition $B$. Since the
accuracy of the partitions varies according to the location of
the 4-tuples in the distribution space, it would seem
advantageous to find a partitioning function whose range has a
constant distribution. The existence of such a function is not
known at present, although one would expect that if such a
function were found, it would not be particularly accurate. In
general, meaningful measurements of human behaviour produce
uneven distributions.

Many alterations made by students to copied programs will be
transparent to this method. Cosmetic transformations such as the
reordering of time-independent statements, recommenting,
reformatting of text, and renaming variables and labels will have
no effect at all on $n_1$, $n_2$, $N_1$, or $N_2$. Most non-cosmetic
alterations fall into one of six well-defined impurity classes,
all of which are detectable by a slightly more sophisticated
counter. Unfortunately, a student who cheated on only part of a
program will not be detected.

Since the parameter counting routine was developed for other
purposes, its $300 or so development cost is not significant
here: its running cost is about five cents ($0.05) per 100 line
student program on a CDC 6500. (This would be less were it not
that the routine was written in ANSI-FORTRAN for portability and
self-analysis.) Thus, this method of detecting plagiarism is
both inexpensive and rapid. The preventive element mentioned in
the title is simply the deterrent created by making it difficult
to cheat successfully.

It seems that this method can not only be applied to programs
in other computer languages, but to any assignment which requires
the submission of written material. Of course, programs are the
only practical item for measurement since they are already in
machine-readable form, but software science has been applied with
some success to English [Kuln 1975, Halstead 1977] and one might
hypothesize that similar results can be obtained there.

ACKNOWLEDGEMENTS

Special notes of thanks are due Dwight Andrews of Purdue
University for collecting copies of his students' programs
expressly for this study and to Professor Halstead, also at
Purdue, for his encouragement and insights into software science.

1 The impurity classes are [Bulut 1974]:
   (1) self-cancelling operations
   (2) ambiguous usage of an operand
   (3) synonymous usages of operands
   (4) common subexpressions
   (5) unnecessary replacements
   (6) unaffected expressions
<table>
<thead>
<tr>
<th>Module</th>
<th>$\lambda_1$</th>
<th>$\lambda_2$</th>
<th>$N_1$</th>
<th>$N_2$</th>
<th>$N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>35</td>
<td>18</td>
<td>34</td>
<td>116</td>
<td>84</td>
<td>200</td>
</tr>
<tr>
<td>14</td>
<td>16</td>
<td>27</td>
<td>121</td>
<td>94</td>
<td>215</td>
</tr>
<tr>
<td>8</td>
<td>15</td>
<td>29</td>
<td>125</td>
<td>93</td>
<td>218</td>
</tr>
<tr>
<td>17</td>
<td>15</td>
<td>33</td>
<td>125</td>
<td>93</td>
<td>218</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>33</td>
<td>131</td>
<td>93</td>
<td>224</td>
</tr>
<tr>
<td>26</td>
<td>15</td>
<td>34</td>
<td>132</td>
<td>96</td>
<td>228</td>
</tr>
<tr>
<td>22</td>
<td>19</td>
<td>32</td>
<td>135</td>
<td>93</td>
<td>228</td>
</tr>
<tr>
<td>31</td>
<td>15</td>
<td>32</td>
<td>132</td>
<td>98</td>
<td>230</td>
</tr>
<tr>
<td>27</td>
<td>18</td>
<td>32</td>
<td>135</td>
<td>95</td>
<td>230</td>
</tr>
<tr>
<td>41</td>
<td>19</td>
<td>32</td>
<td>135</td>
<td>95</td>
<td>230</td>
</tr>
<tr>
<td>45</td>
<td>16</td>
<td>35</td>
<td>133</td>
<td>99</td>
<td>232</td>
</tr>
<tr>
<td>44</td>
<td>17</td>
<td>36</td>
<td>131</td>
<td>101</td>
<td>232</td>
</tr>
<tr>
<td>47</td>
<td>20</td>
<td>33</td>
<td>134</td>
<td>98</td>
<td>232</td>
</tr>
<tr>
<td>11</td>
<td>18</td>
<td>32</td>
<td>129</td>
<td>105</td>
<td>234</td>
</tr>
<tr>
<td>2</td>
<td>17</td>
<td>37</td>
<td>136</td>
<td>101</td>
<td>237</td>
</tr>
<tr>
<td>16</td>
<td>15</td>
<td>36</td>
<td>137</td>
<td>101</td>
<td>238</td>
</tr>
<tr>
<td>39</td>
<td>15</td>
<td>35</td>
<td>138</td>
<td>101</td>
<td>239</td>
</tr>
<tr>
<td>36</td>
<td>17</td>
<td>28</td>
<td>138</td>
<td>103</td>
<td>242</td>
</tr>
<tr>
<td>32</td>
<td>18</td>
<td>35</td>
<td>139</td>
<td>103</td>
<td>242</td>
</tr>
<tr>
<td>40</td>
<td>19</td>
<td>37</td>
<td>141</td>
<td>101</td>
<td>242</td>
</tr>
<tr>
<td>20</td>
<td>17</td>
<td>35</td>
<td>139</td>
<td>105</td>
<td>244</td>
</tr>
<tr>
<td>28</td>
<td>15</td>
<td>37</td>
<td>139</td>
<td>107</td>
<td>246</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>35</td>
<td>138</td>
<td>108</td>
<td>246</td>
</tr>
<tr>
<td>12</td>
<td>15</td>
<td>35</td>
<td>135</td>
<td>111</td>
<td>246</td>
</tr>
<tr>
<td>21</td>
<td>16</td>
<td>35</td>
<td>135</td>
<td>111</td>
<td>246</td>
</tr>
<tr>
<td>13</td>
<td>18</td>
<td>35</td>
<td>142</td>
<td>104</td>
<td>246</td>
</tr>
<tr>
<td>24</td>
<td>15</td>
<td>36</td>
<td>135</td>
<td>113</td>
<td>248</td>
</tr>
<tr>
<td>43</td>
<td>18</td>
<td>34</td>
<td>146</td>
<td>106</td>
<td>252</td>
</tr>
<tr>
<td>6</td>
<td>15</td>
<td>33</td>
<td>143</td>
<td>111</td>
<td>254</td>
</tr>
<tr>
<td>18</td>
<td>15</td>
<td>42</td>
<td>144</td>
<td>112</td>
<td>256</td>
</tr>
<tr>
<td>9</td>
<td>16</td>
<td>35</td>
<td>139</td>
<td>117</td>
<td>256</td>
</tr>
<tr>
<td>4</td>
<td>21</td>
<td>35</td>
<td>150</td>
<td>106</td>
<td>256</td>
</tr>
<tr>
<td>38</td>
<td>16</td>
<td>28</td>
<td>141</td>
<td>117</td>
<td>258</td>
</tr>
<tr>
<td>10</td>
<td>19</td>
<td>35</td>
<td>149</td>
<td>109</td>
<td>258</td>
</tr>
<tr>
<td>33</td>
<td>16</td>
<td>40</td>
<td>143</td>
<td>121</td>
<td>264</td>
</tr>
<tr>
<td>46</td>
<td>18</td>
<td>37</td>
<td>153</td>
<td>120</td>
<td>273</td>
</tr>
<tr>
<td>3</td>
<td>18</td>
<td>41</td>
<td>153</td>
<td>121</td>
<td>274</td>
</tr>
<tr>
<td>18</td>
<td>13</td>
<td>39</td>
<td>151</td>
<td>127</td>
<td>288</td>
</tr>
<tr>
<td>30</td>
<td>20</td>
<td>40</td>
<td>152</td>
<td>126</td>
<td>288</td>
</tr>
<tr>
<td>23</td>
<td>15</td>
<td>36</td>
<td>159</td>
<td>133</td>
<td>291</td>
</tr>
<tr>
<td>29</td>
<td>19</td>
<td>31</td>
<td>160</td>
<td>120</td>
<td>300</td>
</tr>
<tr>
<td>15</td>
<td>14</td>
<td>41</td>
<td>170</td>
<td>142</td>
<td>312</td>
</tr>
<tr>
<td>34</td>
<td>17</td>
<td>38</td>
<td>179</td>
<td>142</td>
<td>321</td>
</tr>
<tr>
<td>7</td>
<td>24</td>
<td>39</td>
<td>182</td>
<td>143</td>
<td>325</td>
</tr>
<tr>
<td>42</td>
<td>15</td>
<td>40</td>
<td>195</td>
<td>147</td>
<td>342</td>
</tr>
<tr>
<td>25</td>
<td>19</td>
<td>45</td>
<td>193</td>
<td>154</td>
<td>347</td>
</tr>
<tr>
<td>37</td>
<td>19</td>
<td>45</td>
<td>193</td>
<td>154</td>
<td>347</td>
</tr>
</tbody>
</table>

Table 1: 47 student program parameter values as partitioned by the software science method (left) and the length function (right).
APPENDIX A

Source Listings of Programs in Partition B
1. N=1
2. IGL=0
3. IF1=0
4. IFS=0
5. IG2=0
6. IF2=0
7. IF2=0
8. C HAVE REFERENCED THE COUNTERS
9. READ 111, KOTS, KFTS, KPTS
10. 111 FORMAT (12, X, 12, X, 12)
11. READ 120, MAX
12. 120 FORMAT (12)
13. C HAVE REFERENCED QUANTITIES ON HAND AND NUMBER
14. READ 100, NUM, ICODE, IBQTS, IBFTS, IBPTS
15. 100 FORMAT (12, X, 14, X, 12, X, 12)
16. C HAVE REFERENCED QUANTITIES AGAINST QUANTITIES ON HAND
17. C 'INSUFFICIENT QUANTITY' RECEIPT PRINTED IF APPLICABLE
18. IF (IBQTS LE KQTS) GO TO 20
19. PRINT 200, NUM, ICODE
20. PRINT 205
21. GO TO 44
22. 20 IF (IBFTS LE KFTS) GO TO 30
23. PRINT 200, NUM, ICODE
24. PRINT 205
25. GO TO 44
26. 30 IF (IBPTS LE KPTS) GO TO 40
27. PRINT 200, NUM, ICODE
28. PRINT 205
29. GO TO 44
30. 40 IF (IBQTS = KQTS) GO TO 20
31. PRINT 200, NUM, ICODE
32. PRINT 205
33. GO TO 44
34. C IF ORDER CAN BE FILLED, COSTS ARE COMPUTED
35. C AND A RECEIPT PRINTED
36. 40 KOTS-KOTS-IBQTS
37. KFTS-KFTS-IBFTS
38. KPTS-KPTS-IBPTS
39. QCBST=KOTS-IBQTS+4.05*FLOAT(IBQTS)
40. FCBST=KFTS-IBFTS+4.15*FLOAT(IBFTS)
41. PCOST=KPTS-IBPTS+2.25*FLOAT(IBPTS)
42. QCBST=QCBST+PCOST+FCBST
43. IF (NUM EQ 1) GO TO 66
44. PRINT 200, NUM, ICODE
45. GO TO 77
46. 66 PRINT 200, NUM, ICODE
47. 77 PRINT 210
48. PRINT 220, IBQTS, QCBST
49. PRINT 230, IBFTS, FCBST
50. PRINT 240, IBPTS, PCOST
51. PRINT 250, QCBST
52. PRINT 300
53. C AFTER THE RECEIPT IS PRINTED, THE COSTS FOR EACH STORE ARE
54. C UPDATED TO BE RECALLED AS A SUMMARY WHEN ALL CARDS ARE READ.
55. C SUMMARY VARIABLES HAVE APPROPRIATE SUFFIXES.
56. C IF (NUM EQ 2) GO TO 32
57. 32 FOR STORE NUMBER 1 AND 2 FOR STORE NUMBER 2.
IF2=IF2+10FTS  
IP2=IP2+10FTS  
QC0S2=6.05*FL0RT(IF2)  
FC0S2=4.15*FL0RT(IF2)  
PC0S2=2.25*FL0RT(IF2)  
G0 TO 44  
N=N+1  
IF(N LE MAX) G0 TO 10  
PRINT 260  
PRINT 210  
PRINT 240. IP1, FC0S1  
PRINT 270. U101  
PRINT 300  
PRINT 290  
PRINT 210  
PRINT 220. IO2, QC0S2  
PRINT 230. IF2, FC0S2  
PRINT 240/ IP2, PC0S2  
PRINT 270/GT0T2  
200 FORMAT C0', 'STORE ', 15X, 'ORDER CODE ', I4)  
201 FORMAT C', 'PRICE', 5X, 'C0ST', 12X, 'IN SUFFICIENT STOCK ON HAND ****')  
220 FORMAT C', 'QUARTS) *6.05 ', F6.2)  
225 FORMAT C', 'FIFTH(S) *4.15 ', F6.2)  
230 FORMAT C', 'PINT(S) *2.25 ', F6.2)  
250 FORMAT C', 'TOTAL $', F7.2)  
270 FORMAT C', 'STORE 1 TOTAL BILL')  
280 FORMAT C', 'STORE 2 TOTAL BILL')  
300 FORMAT C',')  
STOP  
END
### OPERATOR FREQUENCY

<table>
<thead>
<tr>
<th>OPERATOR</th>
<th>FREQUENCY</th>
</tr>
</thead>
<tbody>
<tr>
<td>G. O. S.</td>
<td>40</td>
</tr>
<tr>
<td>+</td>
<td>9</td>
</tr>
<tr>
<td>=</td>
<td>13</td>
</tr>
<tr>
<td>LE</td>
<td>4</td>
</tr>
<tr>
<td>EQ</td>
<td>2</td>
</tr>
<tr>
<td>G010 20</td>
<td>1</td>
</tr>
<tr>
<td>G010 44</td>
<td>4</td>
</tr>
<tr>
<td>G010 30</td>
<td>1</td>
</tr>
<tr>
<td>G010 40</td>
<td>1</td>
</tr>
<tr>
<td>G010 66</td>
<td>1</td>
</tr>
<tr>
<td>G010 77</td>
<td>1</td>
</tr>
<tr>
<td>G010 33</td>
<td>1</td>
</tr>
<tr>
<td>FLAT</td>
<td>9</td>
</tr>
</tbody>
</table>

### OPERAND FREQUENCY

<table>
<thead>
<tr>
<th>OPERAND</th>
<th>FREQUENCY</th>
</tr>
</thead>
<tbody>
<tr>
<td>TB1</td>
<td>1</td>
</tr>
<tr>
<td>KFTS</td>
<td>3</td>
</tr>
<tr>
<td>QCBST</td>
<td>2</td>
</tr>
<tr>
<td>QCBS1</td>
<td>2</td>
</tr>
<tr>
<td>QCBS2</td>
<td>2</td>
</tr>
<tr>
<td>PCBST</td>
<td>2</td>
</tr>
<tr>
<td>PCBS1</td>
<td>2</td>
</tr>
<tr>
<td>PCBS2</td>
<td>2</td>
</tr>
<tr>
<td>INITS</td>
<td>5</td>
</tr>
<tr>
<td>NUM</td>
<td>1</td>
</tr>
<tr>
<td>G0101</td>
<td>1</td>
</tr>
<tr>
<td>G0102</td>
<td>1</td>
</tr>
<tr>
<td>IN0ITS</td>
<td>5</td>
</tr>
<tr>
<td>IN0FTS</td>
<td>5</td>
</tr>
<tr>
<td>IP4</td>
<td>4</td>
</tr>
<tr>
<td>IP2</td>
<td>4</td>
</tr>
<tr>
<td>FCBST</td>
<td>2</td>
</tr>
<tr>
<td>FCBS1</td>
<td>2</td>
</tr>
<tr>
<td>FCBS2</td>
<td>2</td>
</tr>
<tr>
<td>IF1</td>
<td>4</td>
</tr>
<tr>
<td>IF2</td>
<td>4</td>
</tr>
<tr>
<td>KFTS</td>
<td>1</td>
</tr>
<tr>
<td>IQ1</td>
<td>4</td>
</tr>
<tr>
<td>IQ2</td>
<td>4</td>
</tr>
<tr>
<td>MNR</td>
<td>1</td>
</tr>
<tr>
<td>KNS</td>
<td>3</td>
</tr>
<tr>
<td>1.00E+00</td>
<td>4</td>
</tr>
<tr>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>6.00E+00</td>
<td>3</td>
</tr>
<tr>
<td>4.15E+00</td>
<td>3</td>
</tr>
<tr>
<td>2.52E+00</td>
<td>3</td>
</tr>
</tbody>
</table>

**ETA1 = 18**

**N1 = 135**
C

FOLLOWING  CALCULATIONS  ARE  FOR  D. T. D. PERTAINING  TO  WEEKLY  SALES

N=1

IQTS=0

IF IFBO

IQTS=0

IF IFBO IFBO

IF IFBO IFBO

FOLLOWING  DETERMINES  WHETHER  OR  NOT  THE  ORDER  CAN  BE  FILLED.

IF (IQTB. LE, LOTS) GOTO 11

IF THE  ORDER  CANNOT  BE  FILLED,  THIS  INFORMATION  WILL  BE  PRINTED.

PRINT100, NUMST, 10RC0D

PRINT110

GOTO 055

IF (IFIB. LE, LFTS) GOTO 12

PRINT100, NUMST, I0RC0D

PRINT110

GOTO 055

IF (IPTB. LE. LPTS) GOTO 20

PRINT100, NUMST, 10RC0D

PRINT110

GOTO 055

LCIT=IQTS-IQTB

LFTS=LFTS-IFIB

LPTS=LPTS-IPTB

FOLLOWING  DETERMINES  ALL  COST  INFORMATION  IF ORDER  CAN  BE  FILLED.

QCOST=6.05*FLOAT(IQTB)

FCOST=4.15*FLOAT(IFIB)

PCOST=2.25*FLOAT(IPTB)

TOT=QCOST+FCOST+PCOST

PRINT100, NUMST, I0RC0D

GOTO 022

PRINT111

PRINT112, I0TB, QCOST

PRINT113, IFIB, FCOST

PRINT114, IPTB, PCOST

PRINT115, TOT

PRINT119

PRINT101, NUMST. 10RC0D

PRINT110

GOTO 025

I0T=I0T+I0TB

IF1=IF1+IFIB

IF1=IF1+IFIB

OCS=6.05*FLOAT(I0T)

PCOST=4.15*FLOAT(IFIB)

PCOST=2.25*FLOAT(IPTB)

GOTO 005

GOTO 005
59. 25 IQTS=IQTS+IOTB
60. IFIF=IFIF+IFIB
61. IPTS=IPTS+IPTB
62. OCO50=O.05*FLOAT(IQTS)
63. FCOS0=0.15*FLOAT(IFIF)
64. PCOS0=2.25*FLOAT(IPTS)
65. GTS0=OCOS0+FCOS0+PCOS0
66. 59 IFIN.LE.99999999
67. C THIS PRINTS OUT THE TOTAL BILL
68. PRINT116
69. PRINT117
70. PRINT118
71. PRINT119
72. PRINT1110
73. PRINT1111
74. PRINT1112
75. PRINT1113
76. PRINT1114
77. PRINT1115
78. PRINT1116
79. PRINT1117
80. PRINT1118
81. PRINT1119
82. PRINT1120
83. PRINT1121
84. PRINT1122
85. PRINT1123
86. PRINT1124
87. PRINT1125
88. PRINT1126
89. PRINT1127
90. PRINT1128
91. PRINT1129
92. PRINT1130
93. PRINT1131
94. PRINT1132
95. PRINT1133
96. PRINT1134
97. PRINT1135
98. PRINT1136
99. PRINT1137
100. FORMRTC'0', 12X,'STORE', X, 12X,'ORDER CODE', X, 14
101. FORMRTC '0', '**** ORDER NOT FILLED'.
102. 1 'INSUFFICIENT STOCK ON HAND ****'
103. FORMRTC'0', '15X,'STORE 1 TOTAL BILL'
104. FORMRTC'0', '15X,'STORE 2 TOTAL BILL'
105. FORMRTC'0', '15X,'STORE 3 TOTAL BILL'
106. FORMRTC'0', '15X,'STORE 4 TOTAL BILL'
107. FORMRTC'0', '15X,'STORE 5 TOTAL BILL'
108. FORMRTC'0', '15X,'STORE 6 TOTAL BILL'
109. FORMRTC'0', '15X,'STORE 7 TOTAL BILL'
110. FORMRTC'0', '15X,'STORE 8 TOTAL BILL'
111. FORMRTC'0', '15X,'STORE 9 TOTAL BILL'
112. FORMRTC'0', '15X,'STORE 10 TOTAL BILL'
113. FORMRTC'0', '15X,'STORE 11 TOTAL BILL'
114. FORMRTC'0', '15X,'STORE 12 TOTAL BILL'
115. FORMRTC'0', '15X,'STORE 13 TOTAL BILL'
116. FORMRTC'0', '15X,'STORE 14 TOTAL BILL'
117. FORMRTC'0', '15X,'STORE 15 TOTAL BILL'
118. FORMRTC'0', '15X,'STORE 16 TOTAL BILL'
119. FORMRTC'0', '15X,'STORE 17 TOTAL BILL'
120. FORMRTC'0', '15X,'STORE 18 TOTAL BILL'
121. FORMRTC'0', '15X,'STORE 19 TOTAL BILL'
122. FORMRTC'0', '15X,'STORE 20 TOTAL BILL'
123. FORMRTC'0', '15X,'STORE 21 TOTAL BILL'
124. FORMRTC'0', '15X,'STORE 22 TOTAL BILL'
125. FORMRTC'0', '15X,'STORE 23 TOTAL BILL'
126. FORMRTC'0', '15X,'STORE 24 TOTAL BILL'
127. FORMRTC'0', '15X,'STORE 25 TOTAL BILL'
128. FORMRTC'0', '15X,'STORE 26 TOTAL BILL'
129. FORMRTC'0', '15X,'STORE 27 TOTAL BILL'
130. FORMRTC'0', '15X,'STORE 28 TOTAL BILL'
131. FORMRTC'0', '15X,'STORE 29 TOTAL BILL'
132. FORMRTC'0', '15X,'STORE 30 TOTAL BILL'
133. FORMRTC'0', '15X,'STORE 31 TOTAL BILL'
134. FORMRTC'0', '15X,'STORE 32 TOTAL BILL'
135. FORMRTC'0', '15X,'STORE 33 TOTAL BILL'
136. FORMRTC'0', '15X,'STORE 34 TOTAL BILL'
137. FORMRTC'0', '15X,'STORE 35 TOTAL BILL'
138. FORMRTC'0', '15X,'STORE 36 TOTAL BILL'
139. FORMRTC'0', '15X,'STORE 37 TOTAL BILL'
140. FORMRTC'0', '15X,'STORE 38 TOTAL BILL'
141. FORMRTC'0', '15X,'STORE 39 TOTAL BILL'
142. FORMRTC'0', '15X,'STORE 40 TOTAL BILL'
143. FORMRTC'0', '15X,'STORE 41 TOTAL BILL'
144. FORMRTC'0', '15X,'STORE 42 TOTAL BILL'
145. FORMRTC'0', '15X,'STORE 43 TOTAL BILL'
146. FORMRTC'0', '15X,'STORE 44 TOTAL BILL'
147. FORMRTC'0', '15X,'STORE 45 TOTAL BILL'
148. FORMRTC'0', '15X,'STORE 46 TOTAL BILL'
149. FORMRTC'0', '15X,'STORE 47 TOTAL BILL'
150. FORMRTC'0', '15X,'STORE 48 TOTAL BILL'
151. FORMRTC'0', '15X,'STORE 49 TOTAL BILL'
152. FORMRTC'0', '15X,'STORE 50 TOTAL BILL'
153. FORMRTC'0', '15X,'STORE 51 TOTAL BILL'
154. FORMRTC'0', '15X,'STORE 52 TOTAL BILL'
155. FORMRTC'0', '15X,'STORE 53 TOTAL BILL'
156. FORMRTC'0', '15X,'STORE 54 TOTAL BILL'
157. FORMRTC'0', '15X,'STORE 55 TOTAL BILL'
158. FORMRTC'0', '15X,'STORE 56 TOTAL BILL'
159. FORMRTC'0', '15X,'STORE 57 TOTAL BILL'
160. FORMRTC'0', '15X,'STORE 58 TOTAL BILL'
161. FORMRTC'0', '15X,'STORE 59 TOTAL BILL'
162. FORMRTC'0', '15X,'STORE 60 TOTAL BILL'
163. FORMRTC'0', '15X,'STORE 61 TOTAL BILL'
164. FORMRTC'0', '15X,'STORE 62 TOTAL BILL'
165. FORMRTC'0', '15X,'STORE 63 TOTAL BILL'
166. FORMRTC'0', '15X,'STORE 64 TOTAL BILL'
167. FORMRTC'0', '15X,'STORE 65 TOTAL BILL'
168. FORMRTC'0', '15X,'STORE 66 TOTAL BILL'
169. FORMRTC'0', '15X,'STORE 67 TOTAL BILL'
170. FORMRTC'0', '15X,'STORE 68 TOTAL BILL'
171. FORMRTC'0', '15X,'STORE 69 TOTAL BILL'
172. FORMRTC'0', '15X,'STORE 70 TOTAL BILL'
173. FORMRTC'0', '15X,'STORE 71 TOTAL BILL'
174. FORMRTC'0', '15X,'STORE 72 TOTAL BILL'
175. FORMRTC'0', '15X,'STORE 73 TOTAL BILL'
176. FORMRTC'0', '15X,'STORE 74 TOTAL BILL'
177. FORMRTC'0', '15X,'STORE 75 TOTAL BILL'
178. FORMRTC'0', '15X,'STORE 76 TOTAL BILL'
179. FORMRTC'0', '15X,'STORE 77 TOTAL BILL'
180. FORMRTC'0', '15X,'STORE 78 TOTAL BILL'
181. FORMRTC'0', '15X,'STORE 79 TOTAL BILL'
182. FORMRTC'0', '15X,'STORE 80 TOTAL BILL'
183. FORMRTC'0', '15X,'STORE 81 TOTAL BILL'
184. FORMRTC'0', '15X,'STORE 82 TOTAL BILL'
185. FORMRTC'0', '15X,'STORE 83 TOTAL BILL'
186. FORMRTC'0', '15X,'STORE 84 TOTAL BILL'
187. FORMRTC'0', '15X,'STORE 85 TOTAL BILL'
188. FORMRTC'0', '15X,'STORE 86 TOTAL BILL'
189. FORMRTC'0', '15X,'STORE 87 TOTAL BILL'
190. FORMRTC'0', '15X,'STORE 88 TOTAL BILL'
191. FORMRTC'0', '15X,'STORE 89 TOTAL BILL'
192. FORMRTC'0', '15X,'STORE 90 TOTAL BILL'
193. FORMRTC'0', '15X,'STORE 91 TOTAL BILL'
194. FORMRTC'0', '15X,'STORE 92 TOTAL BILL'
195. FORMRTC'0', '15X,'STORE 93 TOTAL BILL'
196. FORMRTC'0', '15X,'STORE 94 TOTAL BILL'
197. FORMRTC'0', '15X,'STORE 95 TOTAL BILL'
198. FORMRTC'0', '15X,'STORE 96 TOTAL BILL'
199. FORMRTC'0', '15X,'STORE 97 TOTAL BILL'
200. FORMRTC'0', '15X,'STORE 98 TOTAL BILL'
201. FORMRTC'0', '15X,'STORE 99 TOTAL BILL'
202. STOP
203. END
### STATISTICS FOR THIS MODULE:

<table>
<thead>
<tr>
<th>OPERATOR</th>
<th>FREQUENCY</th>
</tr>
</thead>
<tbody>
<tr>
<td>LE</td>
<td>4</td>
</tr>
<tr>
<td>EQ</td>
<td>2</td>
</tr>
<tr>
<td>GOTO 11</td>
<td>1</td>
</tr>
<tr>
<td>GOTO 55</td>
<td>4</td>
</tr>
<tr>
<td>GOTO 12</td>
<td>1</td>
</tr>
<tr>
<td>GOTO 20</td>
<td>1</td>
</tr>
<tr>
<td>GOTO 77</td>
<td>1</td>
</tr>
<tr>
<td>GOTO 22</td>
<td>1</td>
</tr>
<tr>
<td>GOTO 25</td>
<td>1</td>
</tr>
<tr>
<td>PRINT 10</td>
<td>1</td>
</tr>
<tr>
<td>FLOAT</td>
<td>9</td>
</tr>
</tbody>
</table>

### OPERAND FREQUENCY

<table>
<thead>
<tr>
<th>OPERAND</th>
<th>FREQUENCY</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOT</td>
<td>1</td>
</tr>
<tr>
<td>N</td>
<td>5</td>
</tr>
<tr>
<td>ODDSOX</td>
<td>0</td>
</tr>
<tr>
<td>LEWEN</td>
<td>2</td>
</tr>
<tr>
<td>FENSO</td>
<td>2</td>
</tr>
<tr>
<td>FEST</td>
<td>2</td>
</tr>
<tr>
<td>LEN</td>
<td>1</td>
</tr>
<tr>
<td>NHX</td>
<td>1</td>
</tr>
<tr>
<td>TOD</td>
<td>5</td>
</tr>
<tr>
<td>DTOTD</td>
<td>1</td>
</tr>
<tr>
<td>FCINS</td>
<td>2</td>
</tr>
<tr>
<td>FCOS</td>
<td>2</td>
</tr>
<tr>
<td>IOT</td>
<td>4</td>
</tr>
<tr>
<td>IFID</td>
<td>5</td>
</tr>
<tr>
<td>DLO</td>
<td>2</td>
</tr>
<tr>
<td>1.00E+00</td>
<td>4</td>
</tr>
<tr>
<td>6.05E+00</td>
<td>2</td>
</tr>
<tr>
<td>4.15E+00</td>
<td>3</td>
</tr>
<tr>
<td>2.25E+00</td>
<td>3</td>
</tr>
</tbody>
</table>

**ETR1= 12**

**N1= 135**

**ETR2= 32**

**N2= 95**
APPENDIX B

Histograms for $\eta_1$, $\eta_2$, $N_1$, and $N_2$
for the Observed Sample
DISTRIBUTION OF N1 VALUES

180.---190. 4
170.---180. 2
160.---170. 4
150.---160. 2
140.---150. 9
130.---140. 21
120.---150. 2
110.---120. 1

0 3 6 9 12 15 18 21 24 27

DISTRIBUTION OF ETA1 VALUES

26.---28. 0
24.---26. 0
22.---24. 1
20.---22. 1
18.---20. 9
16.---18. 13
14.---16. 21
12.---14. 2

0 3 6 9 12 15 18 21
DISTRIBUTION OF N2 VALUES

DISTRIBUTION OF ETA2 VALUES
References


