

# TABLE OF CONTENT

<b>Topic</b>	<b>Page</b>
<b>CHAPTER ONE . . . . .</b>	<b>1</b>
<b>A STRUCTURED APPROACH TO . . . . .</b>	<b>1</b>
<b>PROGRAMMING . . . . .</b>	<b>1</b>
<b>THE INFORMATION TECHNOLOGY REVOLUTION. . . . .</b>	<b>1</b>
<b>LEARNING OBJECTIVES. . . . .</b>	<b>1</b>
<b>INTRODUCTION. . . . .</b>	<b>2</b>
<b>THE INFORMATION TECHNOLOGY REVOLUTION . . . . .</b>	<b>2</b>
<b>THE BUSINESS SYSTEM . . . . .</b>	<b>4</b>
<b>THE SYSTEMS DEVELOPMENT LIFE CYCLE. . . . .</b>	<b>5</b>
<b>THE COMPUTER SYSTEM . . . . .</b>	<b>5</b>
<b>AN EXAMPLE ANALOGUE SIGNAL . . . . .</b>	<b>6</b>
<b>AN EXAMPLE DIGITAL SIGNAL. . . . .</b>	<b>7</b>
<b>WHAT IS DATA?. . . . .</b>	<b>7</b>
<b>PROGRAMMING THE COMPUTER SYSTEM . . . . .</b>	<b>8</b>
<b>STRUCTURE OF A MICROCOMPUTER SYSTEM . . . . .</b>	<b>9</b>
<b>THE MICROPROCESSOR (CPU). . . . .</b>	<b>10</b>
<b>THE PROGRAM INSTRUCTION CYCLE . . . . .</b>	<b>12</b>
<b>THE STORED PROGRAM CONCEPT . . . . .</b>	<b>12</b>
<b>THE ARITHMETIC AND LOGIC UNIT . . . . .</b>	<b>14</b>
<b>HIGH LEVEL PROGRAMMING LANGUAGES . . . . .</b>	<b>16</b>
<b>TRANSLATORS. . . . .</b>	<b>16</b>

<b>THE NEED FOR A STRUCTURED APPROACH.</b> . . . .	<b>.17</b>
<b>THE PROGRAM DEVELOPMENT CYCLE</b> . . . . .	<b>18</b>
<b>REVIEW OF THE PROGRAM SPECIFICATION.</b> . . . . .	<b>18</b>
<b>THE PROGRAM DESIGN</b> . . . . .	<b>18</b>
<b>THE PROGRAM CODING</b> . . . . .	<b>21</b>
<b>THE IMPORTANCE OF PROGRAM REVIEW</b> . . . . .	<b>23</b>
<b>TESTING AND DEBUGGING THE PROGRAM</b> . . . . .	<b>23</b>
<b>DOCUMENTING THE PROGRAM</b> . . . . .	<b>24</b>
<b>QUESTIONS AND EXERCISES</b> . . . . .	<b>25</b>
<b>CHAPTER TWO</b> . . . . .	<b>28</b>
<b>DATA FILE PROCESSING.</b> . . . . .	<b>28</b>
<b>LEARNING OBJECTIVES</b> . . . . .	<b>28</b>
<b>INTRODUCTION</b> . . . . .	<b>29</b>
<b>DATA TYPES.</b> . . . . .	<b>29</b>
<b>DATA STRUCTURES</b> . . . . .	<b>29</b>
<b>RECORDS</b> . . . . .	<b>.30</b>
<b>INPUT/OUTPUT PROGRAMMING</b> . . . . .	<b>.31</b>
<b>INPUT SPECIFICATION.</b> . . . . .	<b>32</b>
<b>OUTPUT SPECIFICATION</b> . . . . .	<b>.33</b>
<b>THE PROGRAM LOGIC DESIGN</b> . . . . .	<b>.33</b>
<b>THE PROGRAM TASKS</b> . . . . .	<b>34</b>
<b>THE PROGRAM FLOWCHART</b> . . . . .	<b>.34</b>
<b>IMPLEMENTATION OF THE PROGRAM CODE.</b> . . . .	<b>44</b>
<b>THE PROGRAM SOURCE CODE</b> . . . . .	<b>.44</b>

THE REMARK STATEMENT. . . . .	46
THE DATA STATEMENT . . . . .	47
THE READ INSTRUCTION AND VARIABLE NAMES . . .	48
THE IF-THEN INSTRUCTION . . . . .	50
THE PRINT INSTRUCTION . . . . .	53
THE GOTO INSTRUCTION . . . . .	55
<b>QUESTIONS AND EXERCISES . . . . .</b>	<b>57</b>
<b>CHAPTER THREE . . . . .</b>	<b>59</b>
<b>ARITHMETIC OPERATIONS IN BASIC. .</b>	<b>59</b>
<b>LEARNING OBJECTIVES . . . . .</b>	<b>59</b>
<b>INTRODUCTION . . . . .</b>	<b>60</b>
<b>ARITHMETIC AND LOGICAL OPERATORS. .</b>	<b>60</b>
<b>THE LET INSTRUCTION . . . . .</b>	<b>63</b>
<b>ADDITION. . . . .</b>	<b>63</b>
<b>SUBTRACTION. . . . .</b>	<b>64</b>
<b>MULTIPLICATION . . . . .</b>	<b>65</b>
<b>DIVISION. . . . .</b>	<b>65</b>
<b>EXPONENTIATION . . . . .</b>	<b>66</b>
<b>MULTIPLE OPERATIONS AND PRECEDENCE . .</b>	<b>67</b>
<b>USING ARITHMETIC RESULTS FROM AN . . . . .</b>	
<b>EXPRESSIONS. . . . .</b>	<b>68</b>
<b>ROUNDING WITH THE INT FUNCTION , , , , ,</b>	<b>68</b>
<b>REPORT EDITING AND PRINT USING FORMATS .</b>	<b>70</b>
<b>THE PRINT USING INSTRUCTION . . . . .</b>	<b>70</b>

<b>EDITING NUMERIC DATA . . . . .</b>	<b>71</b>
<b>EDITING STRING DATA . . . . .</b>	<b>73</b>
<b>EDITING REPORT LINES. . . . .</b>	<b>74</b>
<b>THE TAB FUNCTION . . . . .</b>	<b>76</b>
<b>THE USE OF SEMICOLONS AND COLONS. . . . .</b>	<b>77</b>
<b>THE SAMPLE PROGRAM SPECIFICATIONS . . . . .</b>	<b>78</b>
<b>INPUT DATA FILE . . . . .</b>	<b>78</b>
<b>OUTPUT REPORT . . . . .</b>	<b>79</b>
<b>THE PROGRAM DESIGN . . . . .</b>	<b>79</b>
<b>PROGRAM TASKS . . . . .</b>	<b>80</b>
<b>THE PROGRAM CONTROL LOGIC . . . . .</b>	<b>80</b>
<b>THE PROGRAM SOURCE CODE . . . . .</b>	<b>81</b>
<b>INITIALISATION OF PROGRAM VARIABLES . . . . .</b>	<b>81</b>
<b>PRINTING THE REPORT HEADING . . . . .</b>	<b>82</b>
<b>READING AN INPUT RECORD. . . . .</b>	<b>83</b>
<b>THE DATA PROCESSING LOOP . . . . .</b>	<b>84</b>
<b>PRINTING THE FINAL TOTALS . . . . .</b>	<b>86</b>
<b>LOADING THE PROGRAM DATA FILE . . . . .</b>	<b>86</b>
<b>THE COMPLETE SOURCE PROGRAM . . . . .</b>	<b>87</b>
<b>QUESTIONS AND EXERCISES . . . . .</b>	<b>89</b>

<b>CHAPTER FOUR. . . . .</b>	<b>91</b>
<b>USING THE CASE STRUCTURE FOR . . . . .</b>	
<b>ALTERNATIVE PROCESSING. . . . .</b>	<b>91</b>
<b>LEARNING OBJECTIVES . . . . .</b>	<b>91</b>
<b>INTRODUCTION . . . . .</b>	<b>92</b>
<b>THE SAMPLE PROGRAM IN THIS CHAPTER . . . . .</b>	<b>92</b>
<b>THE COMPARING LOGIC STRUCTURE . . . . .</b>	<b>93</b>
<b>DIFFERENT FORMS OF THE CASE STRUCTURE. . . . .</b>	<b>94</b>
<b>AN EXAMPLE POOR LOGIC STRUCTURE. . . . .</b>	<b>95</b>
<b>CODING THE CASE STRUCTURE IN BASIC. . . . .</b>	<b>97</b>
<b>USING THE RELATIONAL OPERATORS . . . . .</b>	<b>98</b>
<b>HOW NUMERIC DATA VALUES ARE COMPARED. . . . .</b>	<b>99</b>
<b>HOW STRING DATA VALUES ARE COMPARED. . . . .</b>	<b>100</b>
<b>USING IF-THEN TO IMPLEMENT THE CASE . . . . .</b>	
<b>STRUCTURE . . . . .</b>	<b>101</b>
<b>USING IF-THEN-ELSE TO IMPLEMENT THE CASE . . . . .</b>	
<b>STRUCTURE . . . . .</b>	<b>102</b>
<b>THE SAMPLE PROGRAM SPECIFICATION . . . . .</b>	<b>103</b>
<b>THE OUTPUT REPORT . . . . .</b>	<b>103</b>
<b>THE PROGRAM DESIGN . . . . .</b>	<b>104</b>
<b>PROGRAM TASKS . . . . .</b>	<b>104</b>
<b>THE PROGRAM CONTROL LOGIC DESIGN . . . . .</b>	<b>105</b>
<b>THE PROGRAM CODE . . . . .</b>	<b>106</b>

LOADING THE DATA FILE . . . . .	109
THE COMPLETE SOURCE PROGRAM. . . . .	110
QUESTIONS AND EXERCISES . . . . .	112
<b>CHAPTER FIVE . . . . .</b>	<b>114</b>
<b>NESTED CASE STRUCTURES . . . . .</b>	<b>114</b>
<b>LEARNING OBJECTIVES . . . . .</b>	<b>114</b>
<b>INTRODUCTION . . . . .</b>	<b>115</b>
<b>THE NESTED CASE STRUCTURE . . . . .</b>	<b>116</b>
GENERAL CONTROL LOGIC. . . . .	116
AN INCORRECT NESTED CASE STRUCTURE . . . . .	119
CORRECT NESTED CASE STRUCTURE . . . . .	120
<b>CODING THE NESTED CASE STRUCTURE . . . . .</b>	<b>121</b>
<b>RECOGNISING THE NEED FOR A NESTED CASE . . . . .</b>	<b>125</b>
<b>STRUCTURE . . . . .</b>	<b>125</b>
<b>THE LOGICAL OPERATORS. . . . .</b>	<b>126</b>
<b>DEBUGGING FACILITIES IN BASIC. . . . .</b>	<b>129</b>
<b>THE TRON AND TROFF INSTRUCTIONS . . . . .</b>	<b>130</b>
<b>THE STOP AND CONT INSTRUCTIONS . . . . .</b>	<b>131</b>
<b>COMPARING NUMERIC VALUES . . . . .</b>	<b>132</b>
<b>THE SAMPLE PROGRAM SPECIFICATION. . . . .</b>	<b>135</b>
<b>INPUT DATA FILE . . . . .</b>	<b>136</b>
<b>OUTPUT REPORT. . . . .</b>	<b>136</b>
<b>THE PROGRAM LOGIC DESIGN. . . . .</b>	<b>136</b>
<b>PROGRAM TASKS . . . . .</b>	<b>136</b>
<b>THE PROGRAM CONTROL LOGIC . . . . .</b>	<b>137</b>

PROGRAM FLOWCHART . . . . .	137
THE PROGRAM CODE . . . . .	138
LOADING THE DATA FILE . . . . .	141
LISTING OF THE SAMPLE PROGRAM CODE . . . . .	142
<b>QUESTIONS AND EXERCISES . . . . .</b>	<b>144</b>
<b>CHAPTER SIX . . . . .</b>	<b>146</b>
<b>LOOPING AND INTERACTIVE . . . . .</b>	
<b>PROGRAMMING . . . . .</b>	<b>146</b>
<b>LEARNING OBJECTIVES. . . . . , . . . . .</b>	<b>146</b>
<b>INTRODUCTION . . . . .</b>	<b>147</b>
<b>INTERACTIVE PROGRAMMING . . . . .</b>	<b>148</b>
<b>THE INPUT INSTRUCTION . . . . .</b>	<b>149</b>
<b>PROMPTING WITH THE INPUT INSTRUCTION . . . . .</b>	<b>151</b>
<b>PROMPTING WITH THE PRINT INSTRUCTION . . . . .</b>	<b>151</b>
<b>USE OF INPUT WITH MULTIPLE VARIABLE NAMES. . . . .</b>	<b>153</b>
<b>CONTROL LOOPING . . . . .</b>	<b>154</b>
<b>THE FOR-NEXT INSTRUCTION . . . . .</b>	<b>159</b>
<b>VERSIONS OF THE FOR-NEXT LOOP . . . . .</b>	<b>160</b>
<b>FOR-NEXT LOOP SINGLE ENTRY/EXIT POINT . . . . .</b>	
<b>PRINCIPLE . . . . .</b>	<b>162</b>
<b>STEP INCREMENTING . . . . .</b>	<b>164</b>
<b>NEGATIVE STEP INCREMENT . . . . .</b>	<b>166</b>
<b>OMITTING THE STEP COMMAND. . . . .</b>	<b>166</b>
<b>USE OF VARIABLE NAMES IN A FOR INSTRUCTION . . . . .</b>	<b>167</b>

NESTED FOR-NEXT LOOPS. . . . .	168
THE SAMPLE PROGRAM . . . . .	171
THE PROGRAM DESIGN . . . . .	171
PROGRAM TASKS . . . . .	171
THE PROGRAM FLOWCHART. . . . .	171
THE PROGRAM CODE . . . . .	174
VALIDATING THE USER INPUT DATA . . . . .	174
VALIDATING INPUT DATA FOR REASONABLENESS .	175
THE CLEAR SCREEN (CLS) INSTRUCTION . . . . .	177
CODING THE FOR-NEXT LOOP IN THE SAMPLE . . . . .	
PROGRAM . . . . .	177
QUESTIONS AND EXERCISES. . . . .	181
CHAPTER SEVEN . . . . .	183
ARRAYS AND ARRAY PROCESSING. . . . .	183
LEARNING OBJECTIVES . . . . .	183
INTRODUCTION . . . . .	184
CREATING ARRAYS IN BASIC. . . . .	185
LOADING AN ARRAY WITH DATA . . . . .	186
DEFINING MULTIPLE ARRAYS . . . . .	187
THE READ AND THE DATA INSTRUCTIONS . . . . .	189
THE RESTORE INSTRUCTION. . . . .	191
TECHNIQUES FOR LOADING ARRAYS. . . . .	192
SEARCHING FOR DATA VALUES . . . . .	193
ARRAY SEARCH LOGIC . . . . .	195

MISUSE OF THE FOR-NEXT LOOP IN ARRAY SEARCH . . . . .	201
BINARY ARRAY SEARCH . . . . .	202
LOGIC FOR THE BINARY SEARCH. . . . .	204
CODING THE BINARY SEARCH LOGIC IN BASIC . . . . .	207
MULTI-DIMENSIONAL ARRAYS . . . . .	207
LOADING A TWO-DIMENSION ARRAY . . . . .	209
THE MAT FUNCTION . . . . .	210
SEARCHING A MULTI-DIMENSION ARRAY . . . . .	211
THE SAMPLE PROGRAM . . . . .	212
THE PROGRAM DESIGN. . . . .	213
PROGRAM TASKS . . . . .	213
SAMPLE PROGRAM FLOWCHART. . . . .	214
THE PROGRAM CODE . . . . .	215
SAMPLE PROGRAM CODE . . . . .	216
QUESTIONS AND EXERCISES . . . . .	219
CHAPTER EIGHT . . . . .	221
MENUS, ARRAYS, SUBROUTINES AND . . . . .	
SORTING. . . . .	221
LEARNING OBJECTIVES . . . . .	221
INTRODUCTION . . . . .	222
MENUS. . . . .	223
PROCESSING REQUIRED FOR A MENU . . . . .	223
THE MULTIPLE SELECTION CASE STRUCTURE . . . . .	226
IMPLEMENTATION OF THE MULTIPLE SELECTION CASE STRUCTURE WITH THE IF-THEN INSTRUCTION. . . . .	226

<b>THE ON-GOTO INSTRUCTION.</b> . . . . .	228
<b>SUBROUTINES.</b> . . . . .	229
<b>THE GOSUB AND RETURN INSTRUCTIONS</b> . . . . .	230
<b>THE ON-GOSUB INSTRUCTION</b> . . . . .	232
<b>ARRAY PROCESSING.</b> . . . . .	233
LOADING ARRAYS WITH THE INPUT INSTRUCTION.	233
SUMMING NUMERIC VALUES IN ARRAYS . . . . .	235
PRINTING DATA VALUES FROM ARRAYS . . . . .	237
<b>SORTING.</b> . . . . .	239
THE BUBBLE SORT ALGORITHM. . . . .	239
LOGIC OF THE BUBBLE SORT ALGORITHM. . . . .	244
BASIC CODE FOR THE BUBBLE SORT ALGORITHM .	250
<b>USING THE SORT MODULE IN THE SAMPLE.</b> . . . . .	
PROGRAM . . . . .	252
<b>POINTERS TO ARRAY ELEMENTS.</b> . . . . .	254
<b>THE GENERALISED SORT SUBROUTINE.</b> . . . . .	259
<b>THE SAMPLE PROGRAM</b> . . . . .	260
<b>THE PROGRAM DESIGN.</b> . . . . .	261
THE PROGRAM TASKS . . . . .	261
FUNCTIONAL DECOMPOSITION . . . . .	261
THE PROGRAM HIERARCHY CHARTS. . . . .	262
THE PROGRAM FLOWCHARTS . . . . .	268
THE MAIN CONTROL LOGIC . . . . .	269
SUBROUTINES FLOWCHARTS . . . . .	270
<b>THE COMPLETE PROGRAM SOURCE CODE</b> . . . . .	275
<b>QUESTIONS AND EXERCISES</b> . . . . .	279

<b>CHAPTER NINE . . . . .</b>	<b>281</b>
<b>STRING PROCESSING . . . . .</b>	<b>281</b>
<b>LEARNING OBJECTIVES . . . . .</b>	<b>281</b>
<b>INTRODUCTION . . . . .</b>	<b>282</b>
<b>THE SAMPLE PROGRAM . . . . .</b>	<b>283</b>
<b>THE USER INSTRUCTIONS . . . . .</b>	<b>283</b>
<b>CREATING THE PERSONALISED LETTER . . . . .</b>	<b>284</b>
<b>CREATING THE MAILING LABELS . . . . .</b>	<b>286</b>
<b>VALIDATING INPUT DATA . . . . .</b>	<b>286</b>
<b>BASIC STRING FUNCTIONS . . . . .</b>	<b>287</b>
<b>THE INKEY\$ FUNCTION . . . . .</b>	<b>287</b>
<b>THE VAL() FUNCTION . . . . .</b>	<b>291</b>
<b>THE LEN() FUNCTION . . . . .</b>	<b>292</b>
<b>THE STRS() FUNCTION . . . . .</b>	<b>294</b>
<b>THE LEFTS() FUNCTION . . . . .</b>	<b>295</b>
<b>THE RIGHTS() FUNCTION . . . . .</b>	<b>297</b>
<b>THE MIDS() FUNCTION . . . . .</b>	<b>298</b>
<b>VALIDATING WITH THE MIDS() FUNCTION . . . . .</b>	<b>299</b>
<b>SEARCHING WITH THE MIDS() FUNCTION . . . . .</b>	<b>300</b>
<b>SEARCHING FOR A SUB-STRING . . . . .</b>	<b>305</b>
<b>PRINTING THE LETTER WITH THE STRING . . . . .</b>	<b>FUNCTIONS . . . . .</b>
	<b>310</b>
<b>CONCATENATION OF SUBSTRINGS . . . . .</b>	<b>311</b>
<b>MIXING NUMERIC AND STRING DATA. . . . .</b>	<b>313</b>
<b>THE STRING\$() FUNCTION . . . . .</b>	<b>314</b>

<b>THE ASCII CODE SYSTEM.</b> . . . . .	<b>314</b>
<b>THE CHR\$( ) FUNCTION.</b> . . . . .	<b>316</b>
<b>SPECIAL USE OF THE CHR\$( ) FUNCTION</b> . . . . .	<b>317</b>
<b>THE ASC( ) FUNCTION.</b> . . . . .	<b>318</b>
<b>THE SAMPLE PROGRAM DESIGN</b> . . . . .	<b>320</b>
<b>PROGRAM TASKS.</b> . . . . .	<b>320</b>
<b>SUB-MODULES AND HIERARCHY CHARTS</b> . . . . .	<b>321</b>
<b>THE PROGRAM CONTROL LOGIC.</b> . . . . .	<b>328</b>
<b>THE MAIN CONTROL LOGIC</b> . . . . .	<b>328</b>
<b>SUB-ROUTINES FLOWCHARTS</b> . . . . .	<b>329</b>
<b>THE PROGRAM CODING.</b> . . . . .	<b>335</b>
<b>QUESTIONS AND EXERCISES</b> . . . . .	<b>343</b>
<b>CHAPTER TEN.</b> . . . . .	<b>345</b>
<b>MATHEMATICAL PROGRAMMING AND FILE         HANDLING</b> . . . . .	<b>345</b>
<b>LEARNING OBJECTIVES</b> . . . . .	<b>345</b>
<b>INTRODUCTION</b> . . . . .	<b>346</b>
<b>AVAILABLE MATHEMATICAL FUNCTIONS IN                 BASIC.</b> . . . . .	<b>347</b>
<b>REVIEW OF THE LOOP CONTROL STRUCTURE</b> . . . . .	<b>348</b>
<b>ITERATION</b> . . . . .	<b>348</b>
<b>INTEGRATION</b> . . . . .	<b>351</b>
<b>MATHEMATICAL FUNCTIONS AVAILABLE IN                 BASIC</b> . . . . .	<b>354</b>
<b>THE RND( ) FUNCTION</b> . . . . .	<b>354</b>

<b>THE SQR() FUNCTION . . . . .</b>	<b>356</b>
<b>THE SGN() FUNCTION . . . . .</b>	<b>357</b>
<b>THE TRIGONOMETRIC FUNCTIONS. . . . .</b>	<b>358</b>
<b>THE NATURAL LOG FUNCTION. . . . .</b>	<b>363</b>
<b>DEFINING FUNCTIONS WITH THE DEF FN . . . . .</b>	<b>STATEMENT. . 365</b>
<b>ARRAY ALGEBRA. . . . .</b>	<b>366</b>
<b>THE MAT FUNCTION. . . . .</b>	<b>366</b>
<b>MATRICES AND VECTORS . . . . .</b>	<b>367</b>
<b>THE MAT ARITHMETIC INSTRUCTIONS. . . . .</b>	<b>367</b>
<b>FILES AND REPORT GENERATION . . . . .</b>	<b>370</b>
<b>SEQUENTIAL FILES . . . . .</b>	<b>371</b>
<b>STORING DATA ON DISK . . . . .</b>	<b>372</b>
<b>THE OPEN INSTRUCTION. . . . .</b>	<b>373</b>
<b>THE PRINT INSTRUCTION. . . . .</b>	<b>373</b>
<b>STORING STRING DATA. . . . .</b>	<b>374</b>
<b>THE CLOSE INSTRUCTION . . . . .</b>	<b>375</b>
<b>READING DATA FROM A DISK FILE . . . . .</b>	<b>376</b>
<b>THE CONTROL BREAK LOGIC . . . . .</b>	<b>377</b>
<b>QUESTIONS AND EXERCISES . . . . .</b>	<b>380</b>

# INDEX OF TERMS

---

## A

A Structured Approach · 1, 17  
ABS function · 133, 134, 350  
absolute value · 62, 134, 135, 347, 352  
accumulating · 81, 82, 108, 138  
accumulator · 85  
algorithm · *See control logic.*  
alphanumeric characters · 30, 48  
alternative processing · 52, 92-96, 114, 125  
ALU · 9-14  
analogue computer · 6  
analogue signal · 6  
AND · 14, 15, 25  
Arithmetic And Logic Unit · 14. *See ALU*  
Arithmetic And Logical Oper · 60  
arithmetic expression · 63-68, 77, 98, 164, 167, 185, 228, 232, 297-299, 319, 356-358, 368, 369  
Arithmetic Operations · 59  
array · 30  
array search · 187, 193-207, 215, 216, 307-309  
arrays · 29, 30, 183-221, 233-238, 247, 254-259, 297, 366, 367  
Arrows · 20  
ASC() function · 318-320  
ASCII code · 30, 31, 100, 101, 314-320  
Assembler · 16  
assembly language · 8-16  
assignment operator · 61

---

## B

BASIC · 8, 16-39, 45-49, 55-70, 82, 89-104, 112-116, 129-135, 144-187, 199, 201, 207-233, 250, 251, 261, 262, 279-318, 343-372, 380  
binary number · 8, 314, 315  
binary search technique · 183, 202  
binary words · 8-13, 25  
bistable devices · 6  
BIT · 30  
BOOLEAN · 29  
Boolean operators · 14, 98, 114, 115, 144  
bubble sort · 222, 239-243, 252, 254  
buffer · 372-380  
buffer area · 372  
buffer number · 373-375  
Business Information Systems · 4  
Business system · 4

---

## C

case structure · 91-125, 135, 144, 155, 158, 168, 175, 195-199, 226-233, 246, 248, 261, 269, 279, 307, 349  
Characters · 29  
CHR\$( ) function · 316-320  
CLS instruction · 177  
COBOL · 16

Command mode · 132  
common exit point · 122, 227  
comparison operations · 92  
Compilers · 16  
compound statement · 126  
computer program · 7, 8, 25  
Computer System · 5, 8, 17, 25, 26  
Concatenation · 311, 312, 343  
CONNECTOR Symbol · 20  
CONT · 114, 131, 132, 145, 181  
control break · 377-381  
control logic · 20, 84, 94, 124, 171, 215, 268-274, 281  
control variable · 154-162, 178-182, 228-237, 291, 348  
counter · 85  
counter variable · 159-170, 181-187, 202, 234, 237, 257  
counting · 81, 82, 108  
CPU · 8-14, 25-27

---

## D

Data · 7, 9, 25, 26  
data exchange · 246  
data file · 20, 28-56  
Data File Processing · 28  
DATA instruction · 7, 9, 25-29, 47-50, 83, 87, 110, 142, 181-193, 366  
data structure · 29, 30, 58  
data swap · *See data exchange*  
data type · 28-30, 36, 47-50, 57, 74, 75, 150, 253  
data validation · 149, 299  
Debugging · 23  
decimal adjustment · 99  
decision instruction · 39-43, 94, 95, 105, 147, 155-161, 202  
DECISION Symbol · 19  
delimiter · 47, 154, 300-304, 317, 318  
digital computer · 6-12, 27  
digital signal · 6, 7  
DIM instruction · 185, 187, 208, 215  
Direct mode · 132  
discipline approach · 5, 17  
Division · 65

---

## E

editing facilities · 59  
editing format · 71-76, 80-89, 107-109, 138, 237  
editing output data · 70, 81, 89  
Execution Phase · 12  
EXOR · 14, 15  
Exponentiation · 61, 66, 90  
External documentation · 24  
external file · 346

---

## F

Fetch Phase · 12  
field · 30-57  
Firmware · 17  
floating point · 132, 134  
floating point representation · 132  
flowchart · 19-25  
FOR instruction · 159-188, 234  
FOR-NEXT loop · 159-170, 179-191, 201, 202,  
209-219, 233-237, 252-258, 300, 352, 362-368  
FOR-NEXT statement · 147, 159, 170  
functional decomposition · 268

---

## G

GOSUB instruction · 230-233, 252, 253, 291, 292,  
357, 358  
GOTO instruction · 28, 39, 45, 55, 57, 85, 97, 106,  
109, 123, 141, 159, 175-179, 200, 228, 252, 289

---

## H

hexadecimal · 8, 25  
Hexadecimal numbers · 8  
hierarchy chart · 262-269, 280, 320, 327, 328  
High level language · 16  
high level programming · 15-17, 25  
hybrid computer · 7

---

## I

IF-THEN instruction · 28, 37, 45-57, 70, 85, 97-  
144, 158-164, 174-181, 199-201, 219-227, 250-  
253, 288-310, 343, 350, 357  
IF-THEN-ELSE statement · 92, 102, 103, 112  
immediate mode · 131  
indentation · 28, 55  
index array · 254-259, 267  
industrial revolution · 2  
information needs · 4, 25  
Information Technology · 2, 3, 25, 26, *See IT*  
INKEY\$() function · 287-291, 329  
INPUT · 4, 10-19, 31, 32, 78-83, 136, 146-158, 174-  
181, 199, 223-226, 233-238, 286-294, 318, 320,  
363, 364, 373-376  
INPUT instruction · 149-154, 174, 233-238, 376  
INPUT/OUTPUT Symbol · 19  
instruction cycle · 12, 13, 26  
INT() function · 207, 356  
Integer · 29  
integration · 351-354, 380  
interactive programs · 221  
interactive processing · 147, 282  
interactive program · 146-149, 175, 222, 225, 260,  
280, 286  
internal documentation · 283  
interpreter · 16  
IT · 2-5  
IT revolution · 2  
iteration · 348-350, 380  
iterative process · 348

---

## K

key field · 222, 239, 371, 380  
Knowledge Worker · 2

---

## L

LEFT\$() function · 295-297, 310, 311  
LEN() function · 292-297  
LET instruction · 46, 63-67, 76-85, 113, 141, 176,  
191, 215, 250-257, 288-290, 298, 303, 312, 319,  
320, 362-368  
life cycle phases · 5  
List · 29, 58  
LN() function · 363  
loading a set of arrays · 192  
Logic Coder · 104, 215  
Logical operators · 62, 98, 126  
loop counter · *See counter variable.*  
loop logic structure · 60, 93, 146, 154-162, 168,  
177-182, 201, 219, 270, 307, 348-353, 380  
low-level language · 16

---

## M

machine code · 8, 15, 16, 25  
machine language · 8, 13, 25, 26  
machine level · 8-14, 29  
main memory · 9-16, 23, 26, 35, 46-48, 133, 149,  
184-189, 280, 314, 345, 346, 370-376  
maintainability · 17  
MAT function · 210, 345, 366-368, 381, 382  
mathematical · 59  
mathematical functions · 62, 210, 345, 347, 354,  
380  
matrix multiplication · 367, 369  
menu driven programs · 221-223, 282, 283  
menu-driven program · 224  
menus · 221  
microcomputer system · 1, 7, 9, 17  
microprocessor · 8-13, 25  
MID\$() function · 298-313  
mnemonics · 8, 14, 16, 25, 26  
modular design · 320  
Motorola MC6800 · 10, 11  
multi-dimension array · 207-211  
multiple selection structure · 227  
multiplication · 61, 65, 210, 367-369

---

## N

nested case structure · 114-126, 135, 144, 168, 227  
nested FOR-NEXT · 168-170, 209, 210, 366, 368  
nested loop · 246-250, 366  
NEXT instruction · 146, 159-170, 178-181, 187,  
188, 201, 209, 215, 233  
NOT operator · 14, 15  
numeric · 8, 28-30, 36, 47-49, 54-58  
numeric literal · 49, 64, 77, 112, 185, 294, 298, 318,  
357  
Numeric Real · 29

numeric variable · 36, 49, 50, 68, 77, 84, 89, 98, 99,  
112, 152, 159, 181, 185, 227-233, 294, 298, 307,  
318, 320, 343, 356, 357, 365, 374

---

## O

object code · 16  
object program · 16  
off-page connector · 172, 173  
ON-GOTO instruction · 228  
on-page connector · 172  
Op. Code · 9-13  
OPEN instruction · 373-376  
Operand · 9-13  
operating system · 372  
Operational Code · 12. *See Op. Code*  
operators · 60  
OR · 14, 15  
order of precedence · 60  
OUTPUT · 4, 10-19, 31, 33, 53, 79, 103, 130-136,  
151, 165-169, 184, 311-319, 373

---

## P

PAGE CONNECTOR Symbol · 20  
parameter value · 297-299, 316, 355-366  
PC · 11, 12, 13, 14  
place value · 315  
PREDEFINE Symbol · 20  
PRINT instruction · 42, 43, 53-58, 68, 76, 77, 83,  
89, 106, 109, 122, 123, 131, 151-155, 162, 163,  
174, 176, 201, 223, 224, 236, 288-298, 310-320,  
356, 362, 369-376, 382  
PRINT statement · 37, 53-56, 76, 83, 131, 152, 153,  
374  
PRINT USING Formats · 59, 70  
PRINT USING instruction · 70-76, 85-89, 109, 140,  
141, 177, 237, 256, 257, 313  
PROCESSING Symbol · 19  
productivity · 2  
program algorithms · 42, *See Control Logic*  
Program Coding · 18, 21-26  
program control · 17, 20  
program counter · 11-14, 25  
Program design · 18, 26  
program development cycle · 26, 27, 44  
Program Development Cycle · 5, 18, 29  
Program documentation · 18, 24, 26  
Program Instruction Cycle · 12  
program logic design · 18-24, 33, 41-48, 59, 95,  
120, 136  
program maintenance · 64, 81, 108-113, 142  
program menu · 223-234, 260, 264, 283-288  
program review · 23, 25  
Program Source Code · 44, 275  
program specification · 18, 24, 33-38, 138, 139  
Program testing and debugging · 18  
projectiles · 360  
prompt · 148-157, 174-182, 212, 223, 224, 238, 260,  
288  
prompt message · 152, 153, 176  
punctuator · 47

---

## Q

quadratic equation · 356, 357

---

## R

random numbers · 354, 356  
READ instruction · 35-42, 48-58, 83, 84, 105-110,  
139, 141, 155, 156, 167, 186-193, 209, 210, 366,  
368  
real numbers · 133, 134, 144, 145  
record · 20, 29-57  
Register · 9-12, 25  
REM instruction · 28, 46, 58, 106  
RESTORE instruction · 191  
RETURN instruction · 231-238, 301  
Review of program specification · 18  
RIGHT\$( ) function · 297  
RND( ) function · 354-356  
rogue value · 87, 156  
rule of precedence · 60

---

## S

scalar quantities · 367  
scientific notation · 134, 135  
search subscript · 196-206  
secondary storage medium · 346, 370, 381  
Semantics · 29  
sequential file · 371, 373, 380, 381  
sequential search · 204, 206, 219  
Service industry · 2  
service oriented society · 2, 3  
SGN( ) function · 357, 358  
single entry point · 93, 108, 116-120, 155, 162, 163,  
181, 250  
single Entry/Exit point · 94  
single exit point · 91-103, 116-123, 162-164, 170,  
182, 201, 202, 219, 349  
single precision · 134  
sort subroutine · 252-259, 266  
sorting · 221, 222, 239, 243, 245, 255-259  
Specification · 18, 32, 33, 103, 135  
SQR( ) function · 356, 357  
Stack · 29  
step increment · 160, 166, 167, 181, 182  
STEP increment · 164, 165  
STOP · 114, 115, 131, 132, 145  
STR\$( ) function · 259, 294, 313  
string · 28-58  
string constant · 37, 51, 56, 82, 85, 95, 98, 105, 139,  
140, 151-153  
String operators · 60  
string processing · 281, 305, 309  
string variable · 36, 49, 71-84, 98-112, 150, 153,  
187, 237, 291-298, 307, 312, 343, 344, 365, 374  
structured program · 17  
subroutines · 221, 222, 229-233, 252, 253, 259-269,  
280, 322  
subscript · 185-211, 219, 234-258, 297, 367, 368  
sub-string · 101, 154, 296-312, 343  
summation · 348-352, 380  
syntax · 18-25

System Analyst · 4, 5  
system development life cycle · 5  
systematic and discipline approach · 5, 15  
Systems Analysis · 5  
Systems Development Life Cycle · 5

---

## **T**

TAB · 59  
TAB function · 76, 77, 89  
TERMINAL Symbol · 20  
The DIM instruction · 184-187, 209  
The IF-THEN-ELSE structure · 91, 93, 116. *See*  
    *Case Structure*  
The LET Instruction · 63  
The RELATIONAL Operators · 98  
The TAB Function · 76  
THEN statement · 37, 51, 97-102, 112-126, 279  
Timing and Control · 10  
TO statement · 160, 164  
Top down design · 222, 261, 262, 281, 320  
trailer record · 32, 39-43, 51, 85, 87, 139, 147, 156  
translator · 16  
Trees · 29  
trigonometric functions · 62, 358-362, 380  
TROFF · 114, 130, 131, 144  
TRON · 114, 130, 131, 144  
truth value · 14, 15  
two- dimension · *See multi-dimension.*  
array. *See* multi-dimension array.  
array

---

## **U**

understanding · 1, 8, 17-24  
user friendly · 16, 225

---

## **V**

VAL() function · 291, 292, 313  
VAL() function · 291  
validation · 175, 224, 225, 286-300, 360, 361  
variable initialisation · 139, 156  
vector · 210, 367, 381

