Pseudocode for password generation

1. Initialise variables:
   1. Set CAPITALS list to [“A”, “B”, “C”, “D”, “E”]
   2. Set LOWERS list to [“a”, “b”, “c”, “d”, “e”]
   3. Set NUMBERS list to [“1”, “2”, “3”, “4”, “5”]
   4. Set SPECIALS list to [“$”, “&”, “%”]
   5. Set TYPES list to [capitals, lowers, numbers, specials]
   6. Take PASSWORD LENGTH.
2. If the PASSWORD LENGTH is less than 4 GOTO 8
3. Define PASSWORD LIST to store passwords in list format.
4. Define PREFIX as empty, REMAINING LENGTH as PASSWORD LENGTH, and COUNTS as array of Four 0s
5. If remaining length not 0 GOTO step 6
   1. If first character in PREFIX is not from CAPITAL or SMALL set, GOTO 6.5.4.
   2. If any COUNTS is 0, GOTO 6.5.4.
   3. If there are more than two CAPITALS or SPECIALS, GOTO 6.5.4.
   4. Append PASSWORD LIST with PREFIX
   5. GOTO 6.5.4.
6. If INDEX is greater than number of TYPES, GOTO 7
   1. Create a copy of COUNTS array as NEW COUNTS.
   2. Offset NEW COUNTS by INDEX and increment the value by one.
   3. Set TYPE to the offset of TYPES by INDEX.
   4. Increment INDEX by one
   5. If INDEX2 is greater than the number of characters in TYPE, GOTO 6
      1. Append PREFIX with a character by offsetting the TYPE by INDEX2.
      2. Increment INDEX2 by one.
      3. Negate one from REMAINING LENGTH.
      4. GOTO 5
7. GOTO 9
8. Throw value error “Cannot satisfy requirements with less than 4 characters”.
9. Exit Program

Sliding window substring

1. Initialise variables:
   1. Set HEIGHT and WINDOW START to Zero
   2. Set TEXT END to number of characters in text.
   3. Define empty HEIGHT POSITIONS and FREQUENCIES list.
   4. Take user input for K
2. If WINDOW START is bigger than TEXT END, GOTO 3
   1. Define CHARS as empty dictionary.
   2. Define WINDOW POS as WINDOW START
   3. If WINDOW POS is bigger than TEXT END, GOTO 2
   4. If the character at WINDOW POS exists in CHARS, GOTO 2.5
      1. Create a new key value pair using the character at WINDOW POS and 1
      2. GOTO 2.6
   5. Increment the value in CHARS using the character WINDOW POS as a key.
   6. Set MULTIPLES and INDEX to Zero
   7. If INDEX is less than the number of keys in CHARS, GOTO 2.8
      1. Set the current key-value pair to the INDEX of CHARS
      2. Increment INDEX
      3. If the current key-value pair has the value of One, GOTO 2.7
      4. Increment multiples by one
      5. GOTO 2.7
   8. Set the SUBSTRING LENGTH to WINDOW POS – WINDOW START + 1
   9. If MULTIPLES is less than K, GOTO 2.3
   10. If MULTIPLES is greater than K, GOTO 2
   11. If SUBSTRING LENGTH is less than HEIGHT, GOTO 2.3
   12. If SUBSTRING LENGTH is equal to HEIGHT, GOTO 2.12.5
       1. Set HEIGHT to LENGTH
       2. Clear HEIGHT POSITIONS list and append with WINDOW POS
       3. Clear the FREQUENCIES list and append a copy of the CHARS
       4. GOTO 2.3
       5. Append HEIGHT POSITONS with WINDOW POS
       6. Append FREQUENCIES with a copy with CHARS
       7. GOTO 2.3
3. PRINT “Search complete”
4. Exit Program