Solve in any programming language you like. Please provide source code to your solutions.

Puzzle #1

Write a program to accept a nonempty string of 0's and 1's as an argument. The program will print the offsets of runs, each run consisting of all 0's or all 1's, where the runs are longer than 1. For example, if given "0010011" it will print "0, 3, 5" on stdout.

Puzzle #2

Write a program that prints all sequences of 32 digits on stdout, such that each digit is a 0 or 1, each sequence is exactly 32 digits in length, and no sequence has two 1's adjacent in the output. For example, the following sequences should be included in the output (not necessarily in this order):

```
00000000000000000000000000000000
10101010101010100101001010101001
01010101010101010101010101010101
01000000010000000100000001000000
10100100010000100000100000010000
01010010001000010000010000001000
```

The following sequences should not be printed, because each has "11" somewhere in the output:

```
01100000011000000110000001100000
00000000000000000000000000000011
10010010001000010000011000010010
```

The following sequence also should not be printed, because it is not 32 digits in length:

```
0101010000101001000100000100
```

The following sequence also should not be printed, because not every digit is a 0 or 1:

```
AAAA52A9
```

Puzzle #3

Write a program to accept a nonempty string of alphanumeric characters. Define a “run” as a consecutive sequence of a single character. For example, “aaaa” is a run of length 4. The program will print the longest run in the given string. If there is no single longest run, then you may print any of those runs whose length is at least as long as all other runs in the string.

Example input: a
Example output: a

Example input: aab
Example output: aa

Example input: abbbbbcc
Example output: bbbbb

Example input: aabbccdd
Example output: aa

(end)