Lab4: Flappy

One day, you are playing *Flappy* and decide to implement it with **LC-3 assembly language**. In the original game, a bird is flying from left to right, but you may fly from top to bottom. In the game, the bird is represented by 3 continuous letters(for example `aaa`). Without control, it will fall to left, but the user can make it fly to right by 1-9 blocks (chars) by clicking corresponding numbers. By the way, the bird will change its appearance after the user clicked a-z. Here is an example:

```
.....aaa............    User Input(you don't need to print)
....aaa.............
...aaa..............
...........aaa.....    <-9
..........aaa......
................aaa    <-8
..............aaa.
..............aaa..
.............aaa...
............aaa....
...........aaa.....
..........aaa......
.........aaa.......
........aaa........
.......ddd.........    <-d
......ddd..........
.....ddd...........
....ddd............
...ddd.............
..ddd..............
..aaa..............    <-a
.aaa...............
aaa................
aaa................
ooo................    <-o
ooo................
....ooo............    <-4
...ooo.............
```

Requirements

- Write program with **LC-3 assembly language**
- Start your User program at `x3000`
  - Is it the only fragment you need?

- Print 20 chars each line, and use `.` for air
- Falling to gound (the leftmost side) won't end the game
- Flying too high (right) is not allowed. Just put the bird on the rightmost side if it fly out of the screen
- Input only consists of `1 - 9` and `a - z`
- It is recommended to put your interrupt routine on x2000(consider why?)
- Delay for a short time (a loop maybe) between two lines so that our eyes can keep up with the output
- **NO CHEATING**

Grading

Lab 4 takes 7% of the final score, consisting of Check and Report.

Check (50%)

- Contact to your lab TA to check your code. In most cases, it is required to be **OFFLINE**.
- TA will test your code in different cases. Correctness is the primary factor in grading.
- TA will ask you questions to make sure you finish it on your own. It is very important to be familiar with the lab and your code. Suggestion: write some comments in case you forget what your code means.
- You can retry if you fails a check, but there will be a penalty of 10% points in Check part each time.

Report (50%)

- Written in **English**, concise and complete
- Convince TA that you finish the lab on your own
- **No more than** 4 A4 pages
- Consisting of:
    - Algorithm explanation
    - Essential parts of your code with sufficient comments
    - Questions TA asked you and your answer in Check

Other Penalty

- Delay: -10% each day after ddl
- Cheating: -100%, and -10% in final score of the course