

## تشخیص حلقه در لیست پیوندی

برای حل سؤال از فرضی استفاده می‌کنم که توی صورت سؤال به طور مستقیم بهش اشاره نشده ولی میشه از متن نتیجه‌گیری کرد و اون تکراری نبودن اعداد. (اتاق با شماره یکسان یعنی اتاق یکسان). اولین راه حلی که به ذهن آدم می‌رسه ذخیره اعداد توی یک داده ساختاری مثل خود لیست پیوندیه و هر بار که ما دکمه رو فشار می‌دیم، به حلقه روی همه عناصر از قبل ذخیره شده می‌زنیم و اگر عددی برابر عدد جدید پیدا کردیم، لیست حلقه داره ولی اگر عدد تکراری ای نبود عدد جدید رو به لیست پیوندیمون اضافه می‌کنیم و این کار رو تا جایی ادامه می‌دیم که یا عدد تکراری پیدا کنیم و یا عدد جدید null باشه.

این رویکرد از اونجایی که پیچیدگی زمانیش از  $O(n^2)$  و پیچیدگی حافظه‌اش هم  $O(n)$ ، رویکرد خوب و کارآمدی نیست. راه حل بهتر می‌تونه این باشه که به لیست پیوندی خالی تعریف کنیم. بعد از دیدن اولین عدد، اون رو به لیست پیوندیمون اضافه کنیم و همچنین به اشاره‌گر بهش تعریف کنیم. بعد به حلقه می‌زنیم و در هر تکرار دو بار روی دکمه می‌زنیم و این دو عدد جدید رو به لیست پیوندیمون اضافه کنیم و همچنین اشاره‌گرمون رو به گام جلو ببریم. این کار رو تا جایی ادامه می‌دیم که یا عدد جدیدی وجود نداشته باشه و به ما null بده که در این صورت اتاق‌ها تشکیل حلقه نمی‌داده و یا اینکه بعد از اضافه کردن دو عدد جدید، عدد نودی که اشاره‌گر ما بهش اشاره می‌کنه برابر tail لیست پیوندیمون بشه که در این صورت اتاق‌ها تشکیل حلقه می‌دن.

پیچیدگی زمانی و حافظه این راه حل هر دو از  $O(n)$  عه.

//press\_button\_twice → function that presses computer button twice and returns list of two numbers that computer shows.

```
function is_there_loop():
    linked_list = initialize_linked_list() // new empty linked list
    my_ptr = linked_list.head

    while my_ptr is not NULL:
        new_nodes = press_button_twice()

        if new_nodes[1] == NULL:
            return FALSE // no loop found

        linked_list.push(new_nodes[0])
        linked_list.push(new_nodes[1]) // adding new elements

        my_ptr = my_ptr.next //moving my_ptr one step

        if(my_ptr.data == linked_list.tail.data)
            return TRUE //loop found

    return FALSE
```