typedef struct {

struct ListNode\* h;

int n;

} Solution;

/\*\* @param head The linked list's head.

Note that the head is guaranteed to be not null, so it contains at least one node. \*/

Solution\* solutionCreate(struct ListNode\* head) {

Solution\* s = calloc(1, sizeof \*s);

for (s->h = head ; head && ++s->n ; head = head->next);

return s;

}

/\*\* Returns a random node's value. \*/

int solutionGetRandom(Solution\* obj) {

struct ListNode \*t = obj->h;

for (int r = rand() % obj->n ; t && r-- ; t = t->next);

return t->val;

}

void solutionFree(Solution\* obj) {

free(obj);

}

/\*\*

\* Your Solution struct will be instantiated and called as such:

\* Solution\* obj = solutionCreate(head);

\* int param\_1 = solutionGetRandom(obj);

\* solutionFree(obj);

\*/