

به نام خدا



درس طراحی الگوریتم

تمرین سری اول

مدرس درس:

سرکار خانم دکتر ملکی

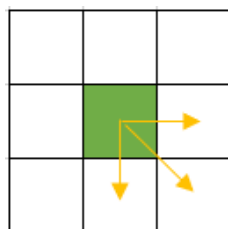
تهیه شده توسط:

الناز رضایی ۹۸۴۱۱۳۸۷

تاریخ ارسال: ۱۴۰۱/۱۲/۰۵

سوال ۱:

لاک‌پشتی می‌خواهد از خانه‌ی آبی به خانه‌ی قرمز برود او در هر حرکت می‌تواند به خانه‌ی راست یا خانه‌ی پایین یا خانه‌ی راست و پایین برود (او از خانه‌ی (j,i) می‌تواند به خانه‌ی $(1+j,i)$ یا $(j,1+i)$ یا $(1+j,1+i)$ برود). هزینه‌ی رفتن به هر خانه در آن نوشته شده است با استفاده از DP به او کمک کنید کم‌هزینه‌ترین مسیر را پیدا کند. پیچیدگی زمانی و حافظه الگوریتم خود را بدست آورده و نحوه محاسبه آنها را توضیح دهید.



مسیر حرکت لاک‌پشت

1	3	3	5	4
5	7	4	4	3
2	3	2	6	2
8	5	6	4	1
3	1	2	5	3

پاسخ ۱:

برای حل این سوال با استفاده از DP، لازم است تا یک جدول با سائز برابر جدول داده شده درست کنیم. سپس، خانه $table(0,0)$ جدول را با هزینه خانه $cost(0,0)$ مقداردهی می‌کنیم. سطر اول جدول $table(0,j)$ را با جمع کردن خانه قبلی در همان سطر $table(0,j-1)$ و هزینه $cost(0,j)$ مقدار دهی می‌کنیم. برای ستون اول نیز، به همین روش عمل می‌کنیم و خانه‌های $table(i,0)$ را از جمع کردن خانه قبلی در همان ستون $table(i-1,0)$ و هزینه همان خانه $cost(i,0)$ به دست می‌آوریم.

$$table(0,0) = 1 \quad , \quad table(0,1) = 4 \quad , \quad table(0,2) = 7 \quad , \quad table(0,3) = 12$$

$$table(0,4) = 16 \quad , \quad table(1,0) = 6 \quad , \quad table(1,2) = 8 \quad , \quad table(1,3) = 16$$

$$table(1,4) = 19$$

1	4	7	12	16
6				
8				
16				
19				

حال برای پر کردن مابقی خانه‌های جدول، مینیمم ۳ خانه دیگری که می‌توانیم از آن‌ها به این خانه برسیم (خانه‌های $table(i-1,j)$ $table(i,j-1)$ $table(i-1,j-1)$)، را به دست آورده و با هزینه آن خانه $cost(i,j)$ جمع می‌کنیم و به همین ترتیب جدول را پر می‌کنیم تا به خانه آخر جدول $table(m,n)$ برسیم. جواب مسئله برابر با $table(m,n)$ می‌باشد.

$$table(1,1) = \min(1, 4, 6) + 7 = 8 \quad , \quad table(1,2) = \min(4, 7, 8) + 4 = 8$$

$$table(1,3) = \min(7, 12, 8) + 4 = 11 \quad , \quad table(1,4) = \min(12, 16, 11) + 3 = 14$$

1	4	7	12	16
6	8	8	11	14
8				
16				
19				

$$table(2,1) = \min(6,8,8) + 3 = 9 \quad , \quad table(2,2) = \min(8,8,9) + 2 = 10$$

$$table(2,3) = \min(8,10,11)+6 = 14 \quad , \quad table(2,4) = \min(11,14,14)+2 = 13$$

1	4	7	12	16
6	8	8	11	14
8	9	10	14	13
16				
19				

$$table(3,1) = \min(8,9,16) + 5 = 13 \quad , \quad table(3,2) = \min(9,10,13) + 6 = 15$$

$$table(3,3) = \min(10,14,15)+4 = 14 \quad , \quad table(3,4) = \min(14,13,14)+1 = 14$$

1	4	7	12	16
6	8	8	11	14
8	9	10	14	13
16	13	15	14	14
19				

$$table(4, 1) = \min(16, 13, 19) + 1 = 14 \quad , \quad table(4, 2) = \min(13, 15, 14) + 2 = 15$$

$$table(4, 3) = \min(15, 14, 15) + 5 = 19 \quad , \quad table(4, 4) = \min(14, 14, 19) + 3 = 17$$

1	4	7	12	16
6	8	8	11	14
8	9	10	14	13
16	13	15	14	14
19	14	15	19	17

هزینه بهترین مسیر، ۱۷ می باشد. همچنین برای یافتن بهترین مسیر نیز کافیست از خانه (۰،۰) جدول شروع کرده و در هر مرحله از بین مسیرهای ممکن، کمترین را انتخاب کرده و به سمت آن حرکت کنیم. بنابراین مسیر انتخابی به شکل زیر در می آید.

1	4	7	12	16
6	8	8	11	14
8	9	10	14	13
16	13	15	14	14
19	14	15	19	17

چون برای پر کردن جدول، به دو حلقه for تودرتو با اندازه‌های m و n نیاز داریم، بنابراین پیچیدگی زمانی از $m * n$ order خواهد بود. همچنین چون سایز جدول $m * n$ است، حافظه نیز از $m * n$ order خواهد بود.