

به نام خدا



تمرین 3 بخش 1

هوش مصنوعی

جناب آقای دکتر رهبان

سارا آذرنوش

98170668

(1

الف)

زندان p = و چاه w = و خروج e =

binary

حالت های در، در میزان باد متفاوت

بدون باد

$$x_i, x_j \in (p, p)$$

کم

$$x_i, x_j \in (p, p), (p, w), (w, p)$$

زیاد

$$x_i, x_j \in (w, w), (w, p), (p, w), (w, e), (e, w)$$

Unary

$$x_1 \in (p, w)$$

$$x_2 \in (p, w)$$

$$x_3 \in (p, w)$$

$$x_4 \in (p, w, e)$$

$$x_5 \in (p, w, e)$$

$$x_6 \in (p, w, e)$$

(ب)

اگر 6 خروج باشد 1 و 5 نیز که درهای مجاور هستند و مقدار باد آنها با 6 زیاد است، چاه هستند در صورتی که بین 1 و 2 باد کم است و اگر 1 چاه باشد باید مقدار باد زیاد باشد. پس 6 چاه است.

(ج)

متغیرهایی که کمترین مقدار را دارند در ابتدا انتخاب میکنیم :

در ابتدا 4 و 6 هستند که هر دو تنها مقدار چاه را دارند و چاه هستند.

سپس در های 1 2 3 نیز زندان یا خروج هستند که خروج نمیتواند در دو در مجاور باشد و چون مقدار 0 نیز نداریم دو زندان نیز در کنار یکدیگر نیستند.

5 نیز بیشترین حالت را دارد.

(د)

5 زندان است و مقادیر مجاور آن زیاد است بنابراین 4 و 6 چاه هستند مقادیر بین 1 2 3 کم هستند بنابراین مقادیر زندان یا خروج دارند که خروج نمیتواند دو در مجاور باشد و دو زندان نیز باعث ایجاد باد نمیشوند که وجود ندارد بنابراین دو حالت زیر برقرار است: (به ترتیب از راست به چپ)

زندان خروج زندان چاه زندان چاه

خروج زندان خروج چاه زندان چاه

(ه)

اگر قیود مسأله با n متغیر را همانند مسأله فوق مطرح کنیم، گراف قیود ما یک گراف دایره‌ای می‌شود که هر گره شامل یک طوقه است. میدانیم با حذف هر گره دلخواه از گراف دایره‌ای میتوانیم آن را تبدیل به گراف بدون دور بکنیم. پس یک گره مانند گره اول که نمایانگر متغیر اول است را از گراف حذف میکنیم تا گراف قیود مسأله تبدیل به یک گراف بدون دور شود. همچنین چون گراف دایره‌ای بود، با این کار تبدیل به درخت نیز میشود. الگوریتم حل به این صورت است که مقدار متغیر اول را که از گراف حذف کرده بودیم یکی از مقادیر دامنه قرار میدهیم. سپس مسأله جدید را که گراف آن را تشکیل دادیم و فاقد متغیر اول است حل میکنیم. حل این مسأله کوچکتر طبق صفحه 34 اسلاید درس، در زمان $O(n \cdot d^2)$ قابل حل است. با توجه به اینکه در ابتدای راه حل باید به تعداد d بار متغیر اول را مقداردهی کنیم و مسأله کوچکتر را اجرا کنیم، زمان نهایی اجرای الگوریتم $O(n \cdot d^3)$ میشود.

(و)

اگر جستجو بکترکینگ انجام دهیم، ارتفاع درخت n است و درجه هر راس d است.

پس در کل d به توان n برگ داریم. اگر آخرین برگ را جواب درست در نظر بگیریم، d^{n-1} بار نیاز به بازگشت خواهیم داشت