

مسئله اول:

۱-۱: رویکرد غیرهوشمندانه برای این مسئله بدین صورت است که تمامی مسیرهای پیمایشی ممکن با در نظر گرفتن حالات انتخابی متفاوت بازیکن‌ها در نظر گرفته شده و توسط الگوریتم Brute-force بررسی شوند.

۱-۲: برای بدست آوردن روش هوشمندانه و بهینه باید به این موضوع دقت کرد که در بعضی از مواقع با توجه به آرایه‌ای از امتیاز مهره‌ها که در اختیار داریم، لزوماً انتخاب مهره با بیشترین امتیاز از دو مهره اول و آخر آرایه منجر به کسب جمع امتیاز بالاتر برای ما نخواهد شد. برای روش بهینه میتوان از رویکرد Dynamic Programming استفاده کرد. المانی را با توجه به شماره دو سر اول و آخر آرایه تحت عنوان حداکثر امتیازی که در یک انتخاب بازیکن اول میتواند کسب کند تعریف میکنیم و همین المان را برای بازیکن دوم به صورت حداقل امتیاز ممکن در انتخابش در نظر می‌گیریم و با همین رویکرد کد برنامه مینویسیم.

۱-۳: اگر بهترین پاسخ‌های ممکن را برای همه المان‌ها از ابتدای آرایه تا پایان آن ذخیره کنیم، حداکثر n^2 زیرمسئله برایمان به وجود می‌آید، لذا پیچیدگی زمانی این الگوریتم از n^2 در n می‌باشد.

۱-۴: پاسخ بهینه در قالب اسکریپت پایتون بارگذاری شده است.

مسئله دوم:

فاصله تبریز و زاهدان را d در نظر میگیریم. اگر باک بنزین پر باشد حداکثر n کیلومتر میتوانیم طی کنیم و فرض میکنیم که سفر از مبدا با باک پر صورت میگیرد. تعداد مکان جایگاه‌های سوخت روی نقشه را هم G در نظر میگیریم.

۱-۲: هنگامی که ما از موقعیت کنونی خود یعنی p مسیرمان را طی میکنیم به شهر قبلی x و y به طور مثال به نام k برمیخوریم، فرقی نمیکند که کدام یکی از دو شهر x و y در موقعیت دورتری نسبت به k قرار گرفته‌اند، انتخاب هر کدام از آنها نسبت به انتخاب k عمل بهتری می‌باشد و لذا پر کردن باک خود در شهر y بدتر از پر کردن باک در شهر x نمی‌باشد.

۲-۲: برای حل این مسئله، الگوریتم حریصانه را پیش می‌گیریم. بدین صورت که یک انتخاب حریصانه انجام می‌دهیم و مسئله را به مسائل کوچکتر تقسیم میکنیم و دوباره وقتی باک بنزین خالی شد همان مراحل را تکرار میکنیم. انتخاب حریصانه به این صورت انجام می‌دهیم که ابتدا در نزدیک‌ترین پمپ بنزین باک را شارژ میکنیم، سپس در دورترین پمپ بنزین قابل دسترس مجدد باک را پر میکنیم و تا وقتی که باک خالی شود ادامه می‌دهیم و هنگامی که باک خالی شد مراحل ذکر شده را مجدد انجام می‌دهیم. اگر راه حل بهینه مطابق با اولین حرکت وجود داشته باشد، انتخاب حریصانه الگوریتمی مناسبی می‌باشد.

مسئله سوم:

رویکردی که برای حل این مسئله در نظر گرفته می‌شود، مبتنی بر تئوری گراف می‌باشد. بدین صورت که میخواهیم با توجه به گراف تشکیل شده از ورودی‌هایمان که تعداد شهرها و مسیرها با فاصله‌شان (وزن هر یال وصل شده بین دو شهر) می‌باشند، کمترین فاصله بین هر دو جفت شهر را بیابیم و در نهایت مجموع این حداقل فواصل بین هر جفت شهر را.

پاسخ در قالب یک اسکریپت پایتون بارگذاری شده است.