جواب میانترم

1 سیستمی که مانند انسان رفتار می کند را با ذکر مثال تشریح کنید؟ یک مثال از یک سیستم هوش مصنوعی که مانند انسان رفتار می کند، سیستم های پردازش زبان طبیعی (NLP) هستند. این سیستم ها به کمک الگوریتم های یادگیری عمیق و پردازش زبان طبیعی، قادرند به صورت خودکار و هوشمند به داده های متنی و گفتاری پاسخ دهند. به عنوان مثال، سیستم های تشخیص سخنرانی می توانند صداهای گفتار را تشخیص داده و به صورت خودکار متن آن را تبدیل کنند. همچنین، سیستم های ترجمه ماشینی نیز قادرند به صورت هوشمند و با استفاده از الگوریتم های یادگیری عمیق، متون را از یک زبان به زبان دیگر ترجمه کنند. این سیستم ها با استفاده از الگوریتم های پردازش زبان طبیعی و یادگیری عمیق، به صورت خودکار و هوشمند با داده های زبانی کار می کنند و رفتار مانند انسان را نشان می دهند. \\\\\\\\\\\\\\\\\\\\\\\\\ 2 هدف از تفکر عاقالنه چیست و چه آورده ای در پی خواهد داشت؟ هدف از تفکر عاقلانه در هوش مصنوعی، ایجاد سیستم هایی است که قادر به تفکر، انتخاب و اتخاذ تصمیمات هوشمندانه با توجه به شرایط محیطی و وضعیت فعلی باشند. این نوع از هوش مصنوعی می تواند به ما در حل مسائل پیچیده، تصمیم گیری های استراتژیک و پیش بینی های دقیق کمک کند. به عنوان مثال، سیستم های تصمیم گیری هوشمند در حوزه مالی می توانند با استفاده از تفکر عاقلانه، به صورت خودکار و هوشمند سرمایه گذاری های موفق را شناسایی کرده و تصمیمات مناسب را اتخاذ کنند. این نوع از هوش مصنوعی، آورده های مهمی در پی خواهد داشت و می تواند به بهبود عملکرد و کارایی در بسیاری از زمینه ها کمک کند. \\\\\\\\\\\\\\\\\\\\\\\\\ 3 اجزای عامل و وظیفه عامل را با رسم شکل و تابع نویسی بررسی کنید؟ در هوش مصنوعی، عامل به معنای هر نوع سیستم هوشمند است که قادر است با محیط ارتباط برقرار کند و به طور خودکار عمل کند. عامل می تواند یک ربات، یک نرم افزار یا حتی یک سیستم هوشمند مبتنی بر وب باشد. اجزای عامل عبارتند از:

حسگرها: حسگرها وظیفه تهیه ورودی های حسی از محیط را دارند. آنها می توانند داده های مانند تصاویر، صداها، دما و فشار را دریافت کنند. حسگرها اطلاعات را به صورت دیجیتال یا آنالوگ به عامل ارائه می دهند. عملگرها: عملگرها وظیفه انجام اقدامات فیزیکی یا محاسباتی را دارند. آنها می توانند موتورها، فعال کننده ها، صفحه نمایش و سایر ابزارهایی باشند که عامل برای تعامل با محیط استفاده می کند. محیط: محیط شامل هر چیزی است که عامل در آن عمل می کند. محیط ممکن است فیزیکی یا مجازی باشد و می تواند شامل اشیاء، سایر عامل ها و سیستم های خارجی نیز باشد. عملکرد: عملکرد عامل، تابعی است که مشخص می کند که عامل در مواجهه با ورودی های خاص چه اقداماتی را انجام دهد. به عبارت دیگر، عملکرد تعیین می کند که عامل چگونه به ورودی ها پاسخ دهد. برنامه: برنامه شامل الگوریتم ها و قوانینی است که عامل برای انجام وظایف خود استفاده می کند. برنامه می تواند به صورت قوانین قابل خواندن توسط انسان یا به صورت کدهای قابل اجرا توسط رایانه باشد. دانش: دانش شامل اطلاعاتی است که عامل از محیط و تجربه های گذشته خود به دست می آورد. دانش می تواند شامل مدل هایی از محیط، قوانین و الگوریتم ها و تجربیات قبلی باشد. به طور کلی، توابع عامل به این صورت می تواند نشان داده شود: def agent\_function(percept): state = update\_state(percept) action = select\_action(state) return action در این تابع، percept ورودی حسگر است که توسط عامل دریافت می شود. تابع update\_state وضعیت عامل را بر اساس percept به روز می کند و تابع select\_action عملیات مورد نیاز برای وضعیت فعلی را انتخاب می کند. سپس عمل انتخاب شده توسط عامل به عنوان خروجی بازگشت داده می شود. \\\\\\\\\\\\\\\\\\\\\\\\\ 4 peas را برای ربات فضانورد و فوتبالیست تشریح کنید؟ PEAS یک مدل است که در هوش مصنوعی به منظور توصیف وظایف و عملکرد عامل‌ها استفاده می‌شود. این مدل شامل چهار عنصر اصلی است که به ترتیب عبارتند از: عملکرد (Performance measure)، محیط (Environment)، عملگرها (Actuators) و حسگرها (Sensors).

عملکرد (Performance measure): این عنصر مشخص می‌کند که عامل چه هدفی را در انجام وظایف خود دنبال می‌کند و عملکرد عامل را ارزیابی می‌کند. برای ربات فضانورد، عملکرد ممکن است شامل انجام مأموریت‌های فضایی مانند بررسی و بررسی سیارات، جمع آوری نمونه‌های سنگی، انجام آزمایشات علمی و حفظ ایمنی در فضا باشد. برای فوتبالیست، عملکرد می‌تواند شامل زدن گل، پاس دادن دقیق، دفاع موثر و کسب برتری در بازی‌ها باشد. محیط (Environment): محیط عبارت است از همه چیزی که عامل در آن عمل می‌کند. برای ربات فضانورد، محیط ممکن است شامل فضا، سیارات، سیستم‌های فضایی و اجسام مختلف دیگری مانند ماهواره‌ها و فضاپیماها باشد. برای فوتبالیست، محیط ممکن است شامل زمین فوتبال، تیم حریف، تماشاگران، توپ و سایر عوامل بازی باشد. عملگرها (Actuators): این عنصر شامل ابزارها و وسایلی است که عامل برای تعامل با محیط استفاده می‌کند. برای ربات فضانورد، عملگرها ممکن است شامل موتورها، دستگاه‌های جمع‌آوری نمونه، دستگاه‌های اندازه‌گیری و دستگاه‌های مخابره باشد. برای فوتبالیست، عملگرها می‌توانند شامل پاها برای زدن گل، دستها برای پاس دادن و دستگاه‌های مخابره با هم‌تیمی‌ها باشند. حسگرها (Sensors): این عنصر شامل حسگرهایی است که عامل برای دریافت اطلاعات از محیط استفاده می‌کند. برای ربات فضانورد، حسگرها ممکن است شامل دوربین‌ها، سنسورهای حرارتی، سنسورهای شیمیایی و سایر حسگرها باشد. برای فوتبالیست، حسگرها می‌توانند شامل چشمان برای دریافت اطلاعات بصری، گوشها برای شنیدن صداها و پوست برای دریافت اطلاعات لمسی باشند. استفاده از مدل PEAS به ما کمک می‌کند تا وظایف و عملکرد ربات فضانورد و فوتبالیست را بهبود بخشیم و سیستم‌های هوشمندی را برای آنها طراحی کنیم. \\\\\\\\\\\\\\\\\\\\\\\\\ 5 طبق شبکه کد زیر چرا عامل مبتنی بر جدول به شکست مواجه می شود؟ راهکارهای پیشنهادی خود را نام برده و مختصری در خصوص هر کدام توضیح دهید؟ عامل مبتنی بر جدول به شکست می‌خورد زیرا در این حالت، عامل فقط بر اساس تاریخچه‌ی مشاهدات قبلی خود عمل می‌کند و هیچ گونه فرایند تصمیم‌گیری یا یادگیری دینامیک ندارد. این در حالی است که در بسیاری از مسائل واقعی، محیط پویا و تغییرپذیر است و عامل باید قادر به انطباق با تغییرات محیط باشد. برای رفع این مشکل و ارتقاء عامل مبتنی بر جدول، می‌توان از راهکارهای زیر استفاده کرد: استفاده از الگوریتم‌های یادگیری تقویتی: الگوریتم‌های یادگیری تقویتی، مانند Q-learning و SARSA، عامل را قادر به یادگیری راه‌حل بهینه برای مسئله می‌کنند. این الگوریتم‌ها بر اساس مفهوم پاداش و تنبیه عمل می‌کنند و با بهبود تصمیم‌گیری عامل، عملکرد آن را بهبود می‌بخشند. استفاده از شبکه‌های عصبی: شبکه‌های عصبی قادر به تعمیم داده‌ها و یادگیری الگوهای پیچیده هستند. با استفاده از این روش، عامل قادر به تصمیم‌گیری در مورد وضعیت‌های مختلف می‌شود و می‌تواند با تغییرات محیط سازگار شود. استفاده از الگوریتم‌های تصمیم‌گیری مارکوف: الگوریتم‌های تصمیم‌گیری مارکوف (MDP) با استفاده از تئوری احتمال و تصمیم‌گیری بهینه، عامل را در تصمیم‌گیری‌های پیچیده راهنمایی می‌کنند. این الگوریتم‌ها مبتنی بر مدلی از محیط هستند و با استفاده از توابع ارزش و عملکرد، به عامل کمک می‌کنند تا راه‌حل بهینه را پیدا کند. با استفاده از این راهکارها، عامل مبتنی بر جدول قادر خواهد بود با محیط پویا و تغییرپذیر تعامل کند و عملکرد بهتری داشته باشد. \\\\\\\\\\\\\\\\\\\\\\\\\ 6دنیای جاروبرقی راباتوجه به فرموله سازی مسئله تشریح کنید؟ حالتها) states ):حالت به وسیله مکان عامل ومکانهای کثیف تعیین میشود.عامل دریکی ازدومکان است که هرکدام ممکن است کثیف باشندیانباشند.پس8=2^ 2\*2 حالت وجوددارد.یعنی)n2.^n)حالت. حالت شروع:هرحالتی میتواندبه عنوان حالت شروع باشد. فعالیتها)action):دراین محیط جاروبرقی فقط سه فعالیت میتواندانجام دهد1:(حرکت به سمت چپ 2(حرکت به سمت راست 3(عمل مکش آزمون هدف:بررسی میکنذآیاتمام مکانهاتمیزاست یاخیر. هزینه ی مسیر:تعدادمراحل موجوددرمسیر،هزینه ی مسیراست. \\\\\\\\\\\\\\\\\\\\\\\\\ 7 جستجوی عمقی را با رسم مرحله به مرحله شرح دهید و در نهایت کارایی الگوریتم را با چهار معیاراندازی گیری بیان کنید؟ جستجوی عمقی (Depth-First Search) یک الگوریتم جستجو در گراف است که با استفاده از استک به صورت عمق به جستجوی گره هدف می‌پردازد. در این الگوریتم، عملیات جستجو از ریشه شروع می‌شود و به صورت پیوسته به سمت پایین درخت جستجو حرکت می‌کند تا به گره هدف برسد یا دیگر قابلیت حرکت نداشته باشد. در هر مرحله، یک گره از استک خارج شده و اعمال ممکن در آن گره بررسی می‌شود. در صورتی که گره هدف یافت شود، جستجو متوقف می‌شود و در غیر این صورت، جستجو به عمق بیشتر ادامه می‌یابد. مراحل جستجوی عمقی به صورت زیر است: قرار دادن گره شروع در استک. تکرار این مراحل تا زمانی که استک خالی شود: گرفتن گره فعلی از استک. بررسی آیا گره فعلی گره هدف است یا خیر. در صورت برابر بودن گره فعلی با گره هدف، جستجو متوقف می‌شود و مسیر به گره هدف برگشت داده می‌شود. در غیر این صورت، تمام فرزندان گره فعلی را به استک اضافه می‌کنیم. در صورتی که استک خالی شود و گره هدف پیدا نشود، جستجو به نتیجه "ناموفق" می‌رسد. کارایی الگوریتم جستجوی عمقی می‌تواند با استفاده از چهار معیار اندازه‌گیری شود: زمان اجرا: زمانی که الگوریتم برای پیدا کردن گره هدف نیاز دارد. حافظه: میزان حافظه مورد نیاز برای ذخیره و مدیریت استک و گره‌های جستجو شده. پیچیدگی زمانی: تعداد گره‌ها و یال‌هایی که باید بررسی شوند تا به گره هدف برسیم. کیفیت جواب: میزان بهینگی جواب‌هایی که الگوریتم جستجوی عمقی به دست می‌دهد. با توجه به معیارهای فوق، الگوریتم جستجوی عمقی معمولاً در مسائلی که گراف جستجو از اندازه بزرگی برخوردار است و تعداد عمق جستجو کم است، کارایی خوبی دارد. اما در مسائلی که عمق جستجو بیشتر است و ممکن است در درخت جستجو به گره‌های تکراری برخورد کند، ممکن است به شکست بخورد و کارایی پایینی داشته باشد. \\\\\\\\\\\\\\\\\\\\\\\\\ 7 جست و جوی عمقی را با رسم مرحله به مرحله شرح دهید و در نهایت کارایی الگوریتم را با چهار معیار اندازه گیری بیان کنید ؟ جستجوی عمقی ،عمیق ترین گره رابسط میدهد،جستجوازعمیق ترین سطح درخت جستجوادامه می یابد،وقتیوقتی گره ها بسط داده شدند ازمرزحذف میشوندوجستجوبه عمیق تری ن گره بعدی برمی گردد.جستجوی عمقی ازصف LIFO استفاده میکند.دراین صف جدیدترین گره تولیدشده،برای بسط دادن انتخاب میشود،این گره بایدعمیق ترین گره بسط نداده شده باشد. جستجوی عمقی: کامل بودن : خیر ، مگر اینکه فضای حالت محدود باشد و حلقه تکرار وجود نداشته باشد . بهینه بودن : خیر ، چون کامل نیست . پیچیدگی زمانی(m^b(O ، اگر m خیلی بزرگتر از d باشد به مراتب بدتر است / در بسیاری از مسائل سریعتر از جست وجوی BF است . پیچیدگی حافظه (+1bm(O : ، در زمان عقبگرد حافظه آزاد می شود . \\\\\\\\\\\\\\\\\\\\\\\\\ 8 ضمن بررسی الگوریتم جستجوی درختی شبه کد زیر را بررسی کنید که استراتژی در کدام از 4 توابع ، پیاده سازی شده است ، توابع را نام برده و عملکرد هر یک را بیان کنید درالگوریتم جستجوی درختی ،حالت شروع درریشه درخت قرارمی گیرد،انشعابها،فعالیتها و گره ها،حالتهای موجودهستند.ابتداریشه رابررسی میکنیم که ایاحالتهدف است یاخیر درصورتی که حالت هدف نبودان رابسط میدهیم تامجموعه ی جدیدی ازحالتهابه وجودآید،بعدازآن حالتهارایکی یکی بررسی کرده تازمانی که به آخرین گره برسیم که هیچ فرزندی ندارد.پس سراغ گره ها میرویم ویکی یکی بررسی میکنیم پس ازآن گره هایی که مارا به هدف نمیرساندحذف میکنیم واین روش ادامه پیدامیکندتابه هدف برسیم. استراتژی های متفاوتی برای رسیدن به حالت هدف وجوددارد.استراتژی مادراینجااین است که یک گره کاندید رابررسی کن اگر هدف نبود آن رابسط بده،آنقدراین کارراتکرار کن تا به هدف برسی. تابع)first remove):اولین خانه را fringeمیکند تابع)test goal):آیابه هدف رسیدیم؟خیر.یک گره باتوجه به استراتژی انتخاب کن تابع)expand):وقتی به هدف نرسیدیم گره هارابسط بده. تابع)insert):گره های فرزندرادرfringبسط بده ونتایج رابه جستجواضافه کن. استراتژی در تابع insert پیاده سازی شده است. \\\\\\\\\\\\\\\\\\\\\\\\\ 9 شبه کد زیر مربوط به کدام جست و جوی ناآگاهانه می باشد ، از مزایای کدام جست و جو های دیگر بهره برده است ، با ترسیم شکل توضیح دهید ؟ این شبه کدمربوط به جستجوی عمقی تکرارشونده است ،که این الگوریتم ازلحاظ زمانی ازمرتبه جستجوی اول سطحی است و ازلحاظ پیچیدگی حافظه ازمرتبه جستجوی اول عمق بهره میبرد. جست و جوی عمقی تکراری ، یک استراتژی کلی است . این الگوریتم با شروع از مقدار صفر به عنوان عمق محدود ، مقدارآن را به تدیج اضافه می کند مانند یک و .. تا ایکه هدفی پیدا شود .

هدف وقتی پیدا می شود که عمق محدود به d برسد ،که d عمق مربوط به عمیق ترین گره هدف است . این الگوریتم از مزایای جست و جوی عمقی و جست وجوی عرضی استفاده می کند فواید مربوط به این دو الگوریتم را با هم ترکیب می کند . این الگوریتم برای تعیین عمق محدود است که جست و جو با عمق محدود را با حدود صعودی تکرار می کند و زمانی خاتمه می یابد که جوابی پیدا شود یا جست و جو با عمق محدود مقدار failure را برگرداند که این عمل نشان می دهد جوابی وجود ندارد . \\\\\\\\\\\\\\\\\\\\\\\\\ 10 شش نوع جست و جو های ناآگاهانه جدول زیر را به تفکیک ، با چهار معیار مربوطه به اختصار شرح دهید ؟ 1(جست و جوی سطحی کامل بودن : بله / شرط : جواب بهینه در عمق d قابل دسترس باشد . فاکتور انشعاب b محدود باشد . بهینه بودن : بله / شرط : مسیر ها فاقد هزینه باشند . پیچیدگی زمانی : گره ریشه حداکثر دارای b فرزند است / هر فرزند نیز حداکثر دارای b فرزند است بنابراین در سطح دوم 2 bگره وجود دارد / با فرض اینکه جواب در عمق d باشد در بدترین حالت جواب باید در سمت راست ترین گره باشد پیچیدگی حافظه : هم مرتبه پیچیدگی زمانی است . 2(جست و جو با هزینه یکنواخت کامل بودن : بله / شرط : جواب در عمق قابل دسترس باشد . هزینه ها مقدار مثبت داشته باشند . بهینه بودن : بله / شرط : کامل باشد . پیچیدگی زمانی : فرض شود c هزینه مسیر بهینه است . فرض شود هزینه هر عمل حداقل e است . در بدترین حالت . است زمانی پیچیدگی( o(b^ce( پیچیدگی حافظه : هم مرتبه پیچیدگی زمانی است . 3(جست و جوی عمق ی کامل بودن : خی ر / شرط : مگر اینکه فضای حالت محدود باشد و حلقه تکرار وجود نداشته باشد . بهینه بودن : خیر / زیرا کامل نیست . پیچیدگی زمانی:) m^ob)است، اگر m خیلی بزرگتر از d باشد به مراتب بدتر است . در بسیاری از مسائل سریعتر از جست و جوی BF است . پیچیدگی حافظه:) +1bm(O در زمان عقبگرد حافظه آزاد می شود . 4(جست و جوی عمقی محدود در حقیقت DF با عمق محدود L است . تعیین در همه مسائل امکان پذیر نمی باشد . اگر d<L آنگاه غیر کامل است . اگر d>L آنگاه کامل اما غیر بهینه است . اگر d=L آنگاه کامل و بهینه است . پیچیدگی زمانی : ) 1b(O پیچیدگی حافظه:(bl(O 5(جست و جوی عمق ی تکراری کامل بودن : بله / شرط : حلقه تکرار وجود نداشته باشد . بهینه بودن : بله / اگر هزینه مسیر ها با هم برابر باشد . O(b^d) : زمانی پیچیدگی پیچیدگی حافظه :(bd(O 6(جست و جوی دو طرف ه کامل بودن : بله / شرط : استفاده کردن از جست و جوی سطری بهینه بودن : بل ه / شرط : استفاده کردن از جست و جوی سطری O(b^d/2):زمانی پیچیدگی O(b^d/2):حافظه پی \\\\\\\\\\\\\\\\\\\\\\\\\ 11 جست وجوی \*A را با توجه به جدول SLD h با جست و جوی حریصانه search Greedy با رسم درختی به طور کامل توضیح داده و تفاوت ها را با دلیل ذکر کنید ؟ دراین روش گره هارا باترکیب (n(g یعنی هزینه رسیدن به گره و (n(h یعنی هزینه رسیدن ازاین گره به گره هدف ارزیابی می کند. (n(h+)n(g)=n(F یعنی (n(f هزینه برآوردشده ی ارزانترین جوار ازطریق n است.پس باید به گره ای فکرکنیم که کمترین (n(gو (n(h راداشته باشد. شناخته شده ترین جستجوی آگاهانه •ایده: از بسط گرههایی که به صرفه به نظر نمیرسند، اجتناب میکند . : f( n)= g( n) +h( n)ارزیابی تابع• (n( g• هزینه واقعی از گره شروع تا گرهn (n( h• هزینه تخمینی از گره n تا هدف (n( f• هزینه تخمینی از گره شروع تا هدف با عبور از گرهn جستجوی \*A کامل وبهینه وبهینه موثراست.مرتبه زمانی ومکانی آن نمایی است.

جستجوی حریصانه:(n( h)= n( f : گره ایی را بسط م یدهد که به هدف نزدیکتر باشد . این جستجوکامل نیست چون حلقه تکراردارد وبهینه هم نیست ومرتبه زمانی ومکانی ان( (m^b(O )است . تفاوت الگوریتم حریصانه \*Aدر(n(g یعنی هزینه واقعی است. Aجستجورابهینه وکامل میکند.جستجوی حریصامه زودتصمیم می گیرد ،امادرA مینیمم ترین گره انتخاب شده وبه آن مینیمم هزینه واقعی اعتمادمیکند. \\\\\\\\\\\\\\\\\\\\\\\\\ 12 الگوریتم زیر را شرح دهید و با توجه به جدول و شکل سوال 11 با رسم درخت جست و جو توضیح دهید ؟ این الگوریتم RBFSاست که درآن : 1(بهترین گره برگ و بهترین جانشین برای آن انتخاب شود . 2(اگر مقدار بهترین گره برگ از جانشین آن بیشتر شد، آنگاه به مسیر جانشین عقبگرد شود . 3(در حین عقبگرد، مقدار (n( f بروزرسانی شود . 4(گره جانشین بسط داده شود . RBFSجستجوی به مراتب موثرتری از \*A ID است . از تولید تعداد بسیار زیادی گره به دلیل تغییر عقیده رنج می برد . مانند \*A اگر (n( h قابل پذیرش باشد، بهینه است . پیچیدگی حافظه ( bd( o است . پیچیدگی زمانی به کیفیت تابع هیوریستیک و میزان تغییر عقیده بستگی دارد . \\\\\\\\\\\\\\\\\\\\\\\\\ 13 چند نوع تابع هیوریستیک را می توان برای پازل اعداد معرفی کرد ، با رسم شکل بررسی کنید ؟ تابع هیوریستیک قابل پذیرش1 •از طریق نسخه ساده شده از مساله(version relax 1h.هر کاشی می تواند به هرجایی منتقل شود 2h ...هر کاشی می تواند به هر خانه همسایه منتقل شود. ABSolver..هزینه راه حل برای مکعب روبیک را تخمین میزند . ابداع تابع هیوریستیک قابل پذیرش (3) •از طریق یادگیر ی از تجربه( experience experience from learning) تجربه : حل تعداد بسیار زیادی از مساله \\\\\\\\\\\\\\\\\\\\\\\\\ 14 سه راه حل جهت ابداع تابع هیوریستیک نام برده و شرح دهید ؟ 1(از طریق نسخه ساده شده از مساله 1 Hهر کاشی می تواند به هر جایی منتقل شود . 2 Hهر کاشی می تواند به هر خانه همسایه منتقل شود . ABSoloverهزینه راه حل برای مکعب روبیک را تخمین می زند . 2(از طریق نسخه کوچکتر از مساله 3(از طریق یادگیری از تجربه تجربه : حل تعداد بسیار زیادی از مساله \\\\\\\\\\\\\\\\\\\\\\\\\ 15 انواع جست و جوی محلی را نام برده و ایده هر یک را بیان کنید ؟ جست و جوی تپه نوردی ، SA، پرتو محلی ، ژنتیک الگوریتم جست و جوی محلی تپه نوردی : این الگوریتم حلقه ای است که در جهت افزایش مقدار حرکت می کند )به طرف باالی تپه ( . وقتی به قله ای رسید که هیچ همسایه ای از آن بلند تر نیست خاتمه می یابد . الگوریتم جست و جوی محلیSA :این الگوریتم نسخه ای از تپه نوردی اتفاقی است و پایین آمدن از تپه مجاز است . حرکت به طرف پایین و به آسانی در اوایل زمانبندی annealing پذیرفته شده و با گذشت طمان کمتر اتفاق می افتد . الگوریتم جست و جوی پرتو محلی : نگهدار ی فقط یک گره در حافظه ، واکنش افراطی نسبت به مسئله محدودیت حافظه است . این الگوریتم به جای بک حالت ، kحالت را نگهداری می کند . این الگوریتم با kحالت که به طور تصادفی تولید شدند ، شرو ع می کند . در هر مرحله تمام پسین های همه حالت ها تولید می شوند . اگر یکی از آن ها هدف بود ، الگوریتم متوقف می شود ؛ وگرنه بهترین پسین را انتخاب و عمل را تکرار می کند . الگوریتم جست و جوی محلی ژنتیک : این الگوریتم شکلی از جست و جوی پرتو اتفاقی است که در آن ، حالت های پسین از طریق ترکیب دو حالت والد تولید می شوند . در مقایسه با انتخاب طبیعی ، مثل جست و جوی پرتو اتفاقی است ، با این تفاوت که اینجا با تولید مثل جنسی سروکار داریم نه غیر جنسی . این الگوریتم همانند جست و جوی پرتو محلی ، با مجموعه ای از k حالت که به طور تصادفی تولید شدند شروع می کند که به آن جعیت گفته می شود \\\\\\\\\\\\\\\\\\\\\\\\\ 16 الگوریتم زیر را شرح داده و انواع آن را نام برده و بررسی کنید ؟ الگوریتم باال مربوط به الگوریتم جست و جوی محلی تپه نوردی می باشد . این الگوریتم حلقه ای است که در جهت افزایش مقدار حرکت می کند ) به طرف باالی تپه( . وقتی به قله ای رسید که هیچ همسایه ای از آن بلند تر نیست خاتمه می یابد . در این الگوریتم درخت جست و جو را نگهداری نمی کند . لذا ساختمان داده گره فعلی فقط باید حالت و مقدار تابع هدف را نگهداری کند . تپه نوردی به همسایه های حالت فعلی نگاه می کند . مثل تالش برای یافتن قله کوه اورست در مه گرفتگی غلیظ ، در حالی که دچار فراموشی هستید . تپه نورد ی گاهی جست و جوی محلی حریصانه نام دارد زیرا بدون اینکه قبال فکر کند به کجا برود ، حالت همسایه خوبی را انتخاب می کند . تپه نوردی معموال به سرعت به جواب پیش می رود ، زیرا به راحتی می تواند حالت بد را بهبود ببخشد .

انواع تپه نوردی ؛ تپه نوردی غیر قطعی : در ب ین حرکت های رو به باال یکی به صورت تصادفی انتخاب می شود . البته احتمال انتخاب با شیب متناسب است . تپه نوردی با انتخاب اولین گزینه : گره ها تا حصول یک گره بهتر بسط داده می شوند . تپه نوردی تصادفی : از حالت شروع مجدد تصادفی تا حصول جواب مجددا شروع خواهد نمود