بسمه تعالي

****

دانشکده مهندسي برق و کامپيوتر

دانشگاه صنعتي اصفهان

1403 - يادگيري ماشين- نيمسال اوّل 1404

1403/7/ تکليف شماره 1 – تحويل شنبه 28

**سپهر عبادی**

9933243

|  |  |
| --- | --- |
|  |  |

1. **تدوین مسئله :**

حالات : اعداد حقیقی مثبت

حالت اولیه : 4

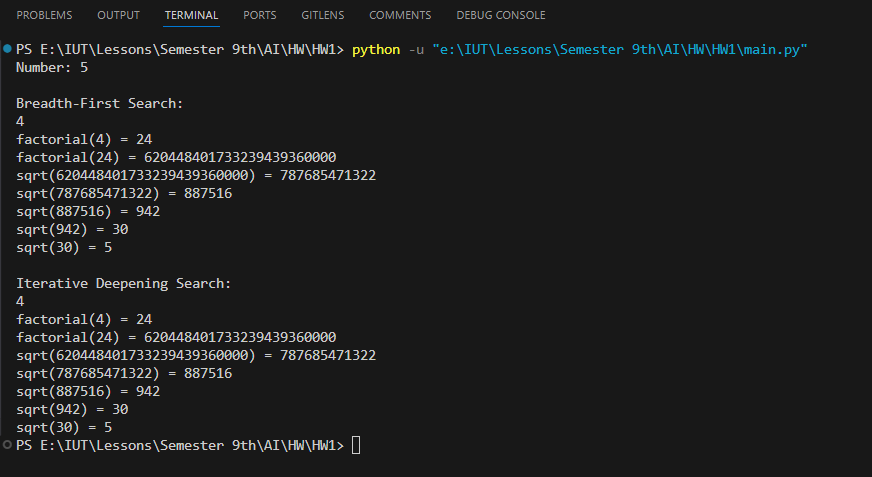
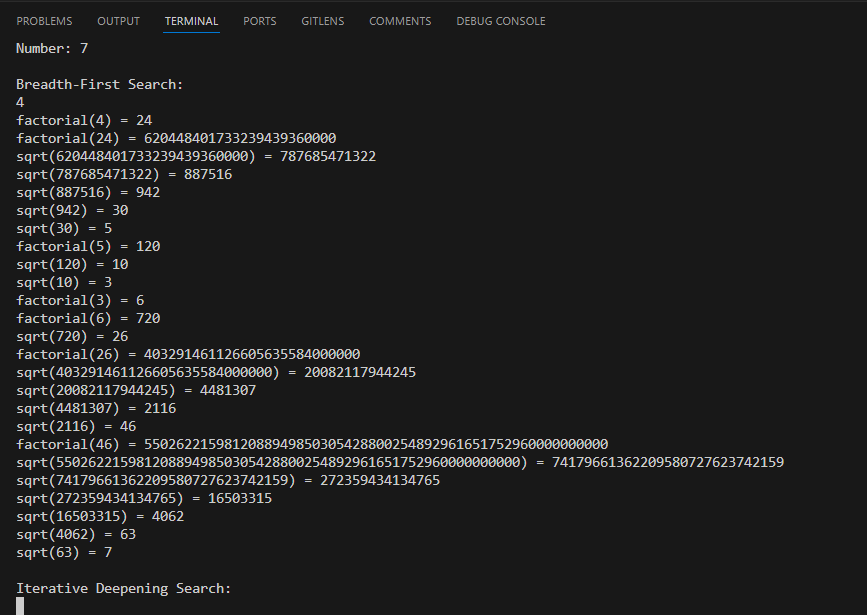
اعمال : جذر، کف و فاکتوریل

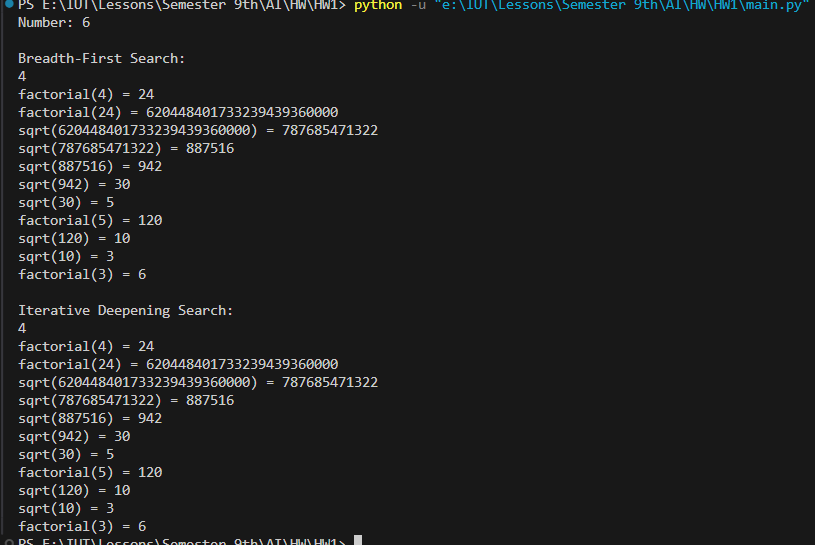
مدل انتقال : با استفاده و ترکیب اعمال موجود باید بتوان از ۴ به آن عدد مورد هدف برسیم؛ که هر عمل مطابق با تعریفشان عمل میکند.

هدف : هر عدد صحیح مثبت مورد انتظار

هزینه : هر عمل ۱

1. **خروجی کد :**





1. **مستندات**

این مسئله باید مسئله knuth را حل کند با استفاده از دو الگوریتم عرض نخست(BFS) و عمیق ساز تکراری(IDS).

اکنون یک توضیح مختصری از هر کدام از این دو الگوریتم جست و جو ارائه میشود :

**BFS** :

الگوریتمی است که به طور سطحی تمام گره‌ها را بررسی می‌کند و به دنبال هدف می‌گردد. این الگوریتم می‌تواند درخت جستجو را به‌صورت کامل و به ترتیب بررسی کند و به همین دلیل برای گره‌های نزدیک به سطح ریشه مناسب است.

**IDS** :

ترکیبی از جستجوی عمق و جستجوی سطح است که با افزایش تدریجی عمق، جستجو می‌کند. این الگوریتم می‌تواند در مواردی که عمق هدف ناشناخته است، مفید باشد.

اکنون توضیحاتی مربوط به پیاده سازی این مسئله که با استفاده از زبان پایتون پیاده سازی شد:

برنامه شامل چهار فایل اصلی است:

1. **knuth.py :** این فایل شامل کلاس KnuthProblem است که منطق مسئله و عملیات‌های ریاضی مجاز را تعریف می‌کند.
2. **bfs.py** : این فایل شامل کلاس BFS است که الگوریتم جستجوی عرض‌نخست (Breadth-First Search) را پیاده‌سازی می‌کند.
3. **ids.py** : این فایل شامل کلاس IDS است که الگوریتم جستجوی عمق تکرارشونده (Iterative Deepening Search) را پیاده‌سازی می‌کند.
4. **main.py** : این فایل اصلی است که در آن از کلاس‌های KnuthProblem، BFS و IDS استفاده می‌شود و ورودی کاربر دریافت می‌شود.

**الگوریتم‌های پیاده‌سازی‌شده**

**جستجوی عرض‌نخست (BFS)**

الگوریتم جستجوی عرض‌نخست یک الگوریتم غیر بازگشتی است که از یک صف FIFO (First-In-First-Out) برای مدیریت گره‌ها استفاده می‌کند. این الگوریتم به‌صورت سطحی جستجو می‌کند و به‌تدریج تمام گره‌های هم‌سطح را بررسی می‌کند. مراحل کلیدی این الگوریتم به صورت زیر است:

گره‌های جدید به صف اضافه می‌شوند تا بررسی شوند.

گره‌ها از صف خارج می‌شوند و بررسی می‌شوند تا ببینند آیا به هدف رسیده‌اند یا خیر.

اگر هدف پیدا شود، مسیر به عنوان خروجی برگردانده می‌شود.

**جستجوی عمیق تکرارشونده (IDS)**

الگوریتم جستجوی عمیق تکرارشونده ترکیبی از جستجوی عمیق و جستجوی سطح است. این الگوریتم از یک تابع داخلی به نام dls (Depth-Limited Search) برای جستجوی عمق محدود استفاده می‌کند. مراحل کلیدی این الگوریتم به صورت زیر است:

در هر بار اجرای dls، عمق فعلی بررسی می‌شود و اگر به هدف برسد، مسیر برگردانده می‌شود.

در صورت رسیدن به عمق محدود، تابع None را برمی‌گرداند.

با افزایش تدریجی عمق، جستجو ادامه پیدا می‌کند تا زمانی که به هدف برسد یا تمام عمق‌های ممکن بررسی شوند.