به نام خدا

|  |  |  |  |
| --- | --- | --- | --- |
| مدرس: مظفر بگ محمدي | دانشگاه ايلام | ترم اول سال تحصيلي 04-03 | سری دوم تمرینات ساختمان داده |

تمرینات کتبی:

1. صف: چگونه مي توان با استفاده از متدهاي استاندارد صف و بدون استفاده از پشته، ترتيب عناصر يک صف را معکوس کرد. براي حل مساله مي توانيد از چندين صف استفاده کنيد.
2. پشته و صف: توضيح دهيد چگونه مي توان عناصر يک پشته را وارد يک پشته ديگر نمود به نحوي که ترتيب عناصر پشته دوم و اول يکسان باشند. مي توانيد از يک صف کمکي براي حل مساله استفاده کنيد.
3. پشته: در مسئله برجهای هانوی، n حلقه که دارای شعاعهای 1 تا n هستند به ترتیب نزولی روی یک میله قرا دارند. دو میله خالی نیز وجود دارند. هدف مسئله انتقال حلقه ها به میله سوم است به طوری که ترتیب حلقه ها تغییر نکند. در ضمن در هر حرکت مجاز به انتقال یک حلقه به میله های دیگر هستید به طوری که هیچگاه یک حلقه بزرگتر روی یک حلقه کوچکتر قرار نگیرد. فرض کنید که شما مجاز به استفاده از یک پشته آرایه ای با اندازه محدود (m) هستید. در این حالت می توانید از هر کدام از میله ها یک حلقه به پشته اضافه کنید و برعکس. توضیح دهید که چگونه می توان با استفاده از این پشته مسئله برجهای هانوی را برای n های بزرگتر از m حل کرد.
4. پشته و صف: تعدادی عنصر در یک پشته قرار دارند. توضیح دهید که چگونه می‌توان کوچکترین عنصر را حذف کرد بدون این که ترتیب بقیه ی عناصر پشته به هم بخورد.
5. **پشته:** فرض کنید N نفر در یک صف قرار دارند. می خواهیم به کمک چندین صف خانمها، آقایان و کودکان را از هم جدا کنیم. به نحوی که در صف خروجی به ترتیب یک کودک، یک زن و یک مرد قرار گرفته باشد. تمام افراد باید در صف خروجی حاضر باشند و ترتیب زنان، کودکان و مردان نیز نسبت به هم دست نخورده باشد.
6. پشته و صف**:** فرض کنید که یک صف و یک پشته داریم. می خواهیم بزرگترین عنصر صف را با کوچکترین عنصر پشته عوض کنیم به نحوی که ترتیب بقیه ی عناصر پشته به هم نخورد. می توانید فقط از یک صف و یک متغییر کمکی نیز استفاده کنید.

تمرینات برنامه نویسی:

1. پشته: با استفاده از پشته بزرگترین مقسوم علیه مشترک دو عدد دلخواه را پیدا کرد.
2. **پشته**: تعدادی عنصر در یک پشته قرار دارند. با استفاده از دو پشته‌ی دیگر عناصر پشته ی اول را مرتب کنید.
3. صف: استفاده از یک پشته عناصر صف ورودی را در صف خروجی مرتب کنید.
4. پشته و صف: با استفاده از دو صف یک پشته درست کنید.
5. پشته: فرض کنید رشته ی s حاوی یک جمله ی انگلیسی است. با استفاده از یک پشته ترتیب کلمات داخل جمله را معکوس کنید. راهنمایی: کاراکترهای s را تا وقتی که به whitespace برنخورده‌‌اید در داخل پشته بریزید، سپس پشته را در خروجی خالی کنید و ادامه دهید. مثال: data structure به atad erutcurts تبدیل می ‌شود.
6. پشته‌: یک رشته از کاربر دریافت کنید و بیشترین تعداد پرانتزهای معتبر این رشته را چاپ کنید.
   * مثلا اگر ورودی به صورت “((ds981((ilam university))))(((((((((data structure)))” باشد تعداد 8 پرانتز معتبر داریم!
7. پشته‌: طول بزرگترین زیردنباله نزولی را در یک آرایه پیدا کنید و زیر دنباله را نمایش دهید.
8. صف: یک متد بنویسید که با استفاده از متدهای استاندارد صف جای عنصر اول و آخر صف را با هم عوض کند. فرض کنید اندازه ی صف بزرگتر یا مساوی دو است. می توانید از یک صف کمکی استفاده کنید.
9. پشته: پشته در ارزیابی عبارات ریاضی بسیار موثر است. معمولا عبارات ریاضی به فرم میانوندی infix داده می شوند و اولویت عملگرهای ریاضی به همراه پرانتز بندی نحوه ی محاسبه ی عبارت ریاضی را تعیین می کند. به عنوان مثال، عبارات ( A + B )\*C و A + ( B \* C ) دارای معانی متفاوتی هستند. در کامپیوتر به جای فرم میانوندی از فرم پسوندی postfix استفاده می شود. به عنوان مثال، معادل پسوندی عبارت (A+B) با AB+ برابر است. محاسبه ی عبارات پسوندی با استفاده از پشته راحت است. لذا، ابتدا باید عبارت میانوندی را به معادل پسوندی آن تبدیل کنیم. سپس، با استفاده از روش زیر عبارت را محاسبه کنیم. دقت کنید که عملگر ^ دارای بالاترین اولویت است. اولویت \* و / از + و – بیشتر است. در شرایط برابر، اولویت عملگرها از چپ به راست تعیین می شود. لذا، معادل پسوندی عبارت (A-B)\*[C/(D+E)+F] با AB- CDE +/F +\* برابر است. الگوریتم ارزیابی یک عبارت پسوندی به صورت زیر است:
10. Push the operands into the stack in the order they are appear.
11. When any operator encounter then pop two topmost operands for executing the operation.
12. After execution push the result obtained into the stack.
13. After the complete execution of expression, the final result remains on the top of the stack.

به عنوان مثال، عبارت میانوندی (2+4) \* (4+6) به عبارت پسوندی 2 4 + 4 6 + \* تبدیل و طبق شکل زیر محاسبه می شود:

