

## Music player

- مسئولین تمرین: امید ناییج نژاد، بهار مرعشی

در این تمرین شما قرار است قابلیت‌های یک برنامه پخش موسیقی را با استفاده از داده‌ساختارهای stack و queue دقت کنید که در این تمرین شما مجاز به استفاده از stack و queue موجود در STL (Standard Template Library) هستید اما لیست پیوندی را باید خودتان پیاده‌سازی کنید.

در هر لحظه یک صفحه با اطلاعات و قابلیت منحصر به فرد خود به کاربر نمایش داده می‌شود و کاربر می‌تواند با ورودی دادن عدد index یا character قابلیت مدنظرش از آن استفاده کند.

همچنین در ابتدای اجرای برنامه، آهنگی در حال پخش نیست اما کاربر به 2 طریق می‌تواند موزیک پخش کند. یا می‌تواند در یک پلی‌لیست گزینه‌ی play را انتخاب کند و یا در صفحه‌ی گزینه‌ی play را انتخاب کند.

از لحظه‌ای که موزیکی شروع به پخش کند باید در هر صفحه‌ای که باشیم در ابتدای آهنگ در حال پخش را نشان دهیم و همینطور قابلیت Skip آهنگ را به کاربر بدهیم.

مثال:

```
song playing: "name of song"
source: "name of the playlist" or "queue"
>. next
<. prev
```

کاربر با ورودی دادن < می‌تواند به آهنگ بعدی برود که با توجه به اینکه در حال پخش از پلی‌لیست است یا صفحه‌ی پخش به روش متفاوتی باید handle شود . اگر در حال پخش از صفحه‌ی next باشد فقط next باید فعال باشد و نمایش داده شود، اگر در حال پخش از پلی‌لیست باشد باید هردو نمایش داده شوند.

از آنجا که در این تمرین، مفهوم زمان در نظر گرفته نمی‌شود، فرض می‌گیریم که آهنگ در حال اجرا ثابت می‌ماند

تا زمانی که دستور پخش دیگری از پلی‌لیست یا صفحه بباید.

در ابتدای اجرای برنامه، پنل اصلی نمایش داده می‌شود:

#### ▼ صفحه اصلی

این صفحه شامل 4 گزینه است

- main page
- 1. songs
- 2. playlists
- 3. song queue
- 0. exit

در صورت انتخاب شدن هر کدام از 3 گزینه‌ی اول، برنامه باید وارد صفحه اختصاصی آن شود که در ادامه به توضیح هرکدام خواهیم پرداخت. و دستور آخر هم به معنای پایان برنامه است.

#### ▼ آهنگ‌ها

اولین گزینه لیست خود آهنگ‌ها هست که باید در صورت نیاز به کاربر نمایش داده شوند. پیاده سازی این لیست دلخواه است اما پیشنهاد می‌شود از `vector` استفاده کنید. همچنین برای هر موزیک هم می‌توانید `Struct` یا `class` تعریف کنید. تصور کنید که در برنامه پخش موزیک آفلاین شما در بخشی می‌توانید تمام موزیک‌های دانلود شده روی دستگاه خود را به ترتیبی برای مثال زمان دانلود مشاهده می‌کنید.

برای مثال:

- list of songs
- 1. track 10, artist B
- 2. track 2, artist D
- 3. track 5, artist A
- 4. track 1, artist C
- 5. track 8, artist E
- 0. back

آخرین خط مربوط به قابلیت برگشت هست که در تمامی پنل‌ها به جز پنل اصلی (در پنل اصلی گزینه‌ی `exit`

وجود دارد) باید باشد.

حال کاربر باید بتواند در این لیست نمایش داده شده با دادن index مورد نظر موزیکی انتخاب کند و وارد صفحه‌ی جزئیات و قابلیت‌های آن بشود.

برای هر آهنگ باید قابلیت‌های زیر وجود داشته باشد:

### نمایش لیست playlist‌های دارای آهنگ مورد نظر

نمایش لیستی از پلی‌لیست‌هایی که تا این لحظه موزیک مورد نظر به آن‌ها اضافه شده است. با انتخاب هر کدام از پلی‌لیست‌ها باید بتوانیم وارد صفحه‌ی آن پلی‌لیست بشویم.

### اضافه کردن به پلی‌لیست

در اینجا کاربر با انتخاب این گزینه و وارد کردن نام پلی‌لیست در صورتی که این موزیک در آن پلی‌لیست از قبل وجود داشته باشد، پیام "song already exists" می‌گیرد و در غیر این صورت لیستی از ترتیب موزیک‌های موجود در آن پلی‌لیست را همراه با سوال "where do you want to add it" دریافت می‌کند که در ادامه، کاربر باید عددی مثل  $x$  ورودی دهد تا آهنگ بعد از جایگاه  $x$  ام اضافه شود. اگر  $x$  بزرگتر از طول پلی‌لیست بود آهنگ به انتهای پلی‌لیست اضافه می‌شود و اگر  $x$  کمتر از 1 بود آهنگ به ابتدای پلی‌لیست اضافه می‌شود. اما لازم است درست بودن فرمت  $x$  (عدد صحیح بودن آن) بررسی شود و در صورت داشتن فرمت اشتباه، پیغام خطای "نمایش داده شود و دوباره ورودی گرفته شود.

زمانی که ورودی قابل قبولی داده شد باید یک پیغام تایید "song added to playlist" نمایش داده شود و سپس برنامه به صفحه‌ی قبل بازگردانده شود.

### اضافه کردن آهنگ به صف پخش

در اینجا صرفا آهنگ به صف پخش اضافه می‌شود و نیازی به چک کردن وجودش از قبل نیست چون امکان پخش چندباره آهنگ در صف پخش وجود دارد و همچنین نیاز به گرفتن جایگاه هم نیست زیرا آهنگ باید به انتهای صف اضافه شود. تنها کافی است یک پیغام جهت تایید انجام شدن دستور به صورت "song added to queue" نمایش داده شود. سپس به صفحه‌ی لیست آهنگ‌ها باز می‌گردد.

### برگشت به صفحه‌ی قبلی

به صفحه‌ی پیشین باز می‌گردیم.

مثال:

```
song name
1. list of playlists
2. add to play list
3. add to queue
0. back
```

#### ▼ پلی‌لیست‌ها

دومین گزینه‌ی صفحه‌ی اصلی، نمایش لیستی از پلی‌لیست‌های موجود است. در این صفحه کاربر می‌تواند یکی از پلی‌لیست‌ها را انتخاب کند و وارد آن شود و یا از قابلیت‌های همین صفحه استفاده کند. پیاده‌سازی این لیست هم دلخواه است.

ابتدا قابلیت‌های صفحه را توضیح می‌دهیم و سپس مثالی از فرمتی که باید به کاربر نمایش داده شود ارائه خواهد شد.

#### ساخت پلی‌لیست جدید

با انتخاب این قابلیت پیغام "enter a name for the new playlist" به کاربر می‌دهیم و چون اسم پلی‌لیست‌ها باید unique باشد اگر اسم تکراری بود پیغام خطای "playlist already exists" را نشان می‌دهد و دوباره اسم را از ورودی می‌گیرد. سپس زمانی که پلی‌لیست جدید ساخته شد، پیغام "new playlist added to" را نمایش می‌دهد و به صفحه‌ی لیست پلی‌لیست‌ها بر می‌گردد و لیست جدید را که در انتهای آن پلی‌لیست تازه اضافه شده قرار دارد را نمایش می‌دهیم.

#### بازگشت به صفحه‌ی قبلی

به صفحه‌ی پیشین باز می‌گردد.

مثال:

### list of playlists

- 1. playlist x
- 2. playlist y
- 3. playlist z
- 1. add playlist
- 0. back

کاربر با انتخاب عدد `index` یک پلی‌لیست می‌تواند وارد صفحه‌ی آن پلی‌لیست شود. که در آن جا ابتدا لیست آهنگ‌های موجود در پلی‌لیست و سپس قابلیت‌های صفحه را می‌بیند.

هر پلی‌لیست یک circular double linked list است به این معنی که اگر یک پلی‌لیست  $n$  آهنگ داشته باشد و آهنگ  $i$  ام از پلی‌لیست در حال اجرا باشد باید بعد از آن آهنگ  $1 + i$  و ... تا آخرین آهنگ (یعنی آهنگ  $n$  ام) اجرا شود و بعد از آن دوباره آهنگ اول و دوم و ... پخش می‌شوند. پخش یک پلی‌لیست زمانی متوقف می‌شود که تمام آهنگ‌ها در آن دور اجرا حداقل یک بار پخش شده باشند. برای مثال برای یک پلی‌لیست دارای 4 آهنگ که از آهنگ دوم شروع شده است داریم: ابتدا آهنگ 2، بعد 3 بعد 4 بعد 1 و سپس پخش پلی‌لیست به پایان می‌رسد.

### قابلیت‌های پلی‌لیست:

کاربر با انتخاب آهنگ مدنظرش می‌تواند وارد صفحه‌ی آن آهنگ شود.

### شروع پخش پلی‌لیست

صرف نظر از اینکه آهنگی از پلی‌لیست دیگری و یا از صفحه پخش در حال اجرا بوده یا نه بعد از انتخاب این دستور یک عدد  $x$  به عنوان ورودی ممکن است و شروع به پخش پلی‌لیست از آهنگ  $x$  ام می‌کند. در اینجا هم اگر  $x$  کمتر از یک بود از آهنگ اول و اگر بیشتر از طول پلی‌لیست بود از آهنگ آخر شروع می‌کند.

اما لازم است درست بودن فرمات  $x$  (عدد صحیح بودن آن) بررسی شود و در صورت داشتن فرمات اشتباه، پیغام خطای "enter an integer" نمایش داده شود و دوباره ورودی گرفته شود.

### پاک کردن آهنگی از پلی‌لیست

با انتخاب این گزینه باید آهنگ مورد نظر از کاربر به صورت "which song" پرسیده شود و همانند

اضافه کردن آهنگ به پلی‌لیست باید چک شود که فرمت عدد ورودی معتبر باشد (اگر عدد وارد شده کمتر از 1 بود آهنگ اول پاک شود و اگر عدد وارد شده بیشتر از طول پلی‌لیست بود آهنگ آخر پاک شود) و در صورت داشتن فرمت اشتباه، پیغام خطای "enter an integer" نمایش داده شود و دوباره ورودی گرفته شود تا زمانی که عدد معتبری وارد شود و بعد از حذف آهنگ پیغام "song deleted" نمایش داده شود. توجه داشته باشید که با پاک کردن موزیک از پلی‌لیست باید لیست پلی‌لیست‌هایی که موزیک در آن‌ها هست هم update شود.

### بازگشت به صفحه‌ی قبل

به صفحه‌ی پیشین باز می‌گردد.

مثال:

```
playlist name
1. track 8, artist E
2. track 1, artist C
-2. play
-1. delete a song
0. back
```

### ▼ صفحه‌پخش

صف پخش به صورت یک queue هست به این معنی که زمانی شروع به پخش شود ابتدا آهنگ سر صف پخش می‌شود سپس آن آهنگ از queue پاک می‌شود و آهنگ دیگری که سر صف قرار گرفته اجرا می‌شود و پخش آهنگ‌ها به همین صورت ادامه می‌یابد تا زمانی که صف خالی شود آنگاه پخش آهنگ‌ها متوقف می‌شود (بر عکس پلی‌لیست که از ابتدا تکرار می‌شود).

در این صفحه فقط کافی است لیست آهنگ‌های موجود در صف پخش تا این لحظه را چاپ کنید به همراه دو قابلیت:

کاربر می‌تواند با انتخاب index آهنگ مورد نظرش وارد صفحه‌ی آن آهنگ شود.

### پخش صف بازگشت به صفحه‌ی پیشین

مثال:

```
songs in queue
1. track 2, artist D
2. track 8, artist E
-1. play
0. back
```

همانطور که در طول سوال دیدید کاربر می تواند در حین پخش یک پلی‌لیست دستور پخش از صف پخش یا پلی‌لیست دیگری را بدهد.

حال فرض کنید که پلی‌لیست x در حال پخش باشد و کاربر دستور پخش پلی‌لیست y را بدهد و در اواسط آن دستور پخش از صف پخش را بدهد چطور باید ترتیب پخش را مدیریت کرد؟

چون نمی‌خواهیم x و y نیمه‌کاره رها شوند زمانی که صف پخش خالی شد ادامه‌ی y را پخش می‌کنیم البته که باید مراقب باشیم در این وقفه‌ی پیش آمده ممکن است تغییراتی در ساختار پلی‌لیست y اتفاق افتد!! تا زمانی که پخش پلی‌لیست y تمام شود (در بالا تمام شدن پخش پلی‌لیست را توضیح داده بودیم) و بعد به ادامه‌ی پخش پلی‌لیست x باز می‌گردد تا پخش پلی‌لیست x هم تمام شود.

مثال:

فرض کنید x دارای 6 آهنگ است و از آهنگ یک شروع به پخش شده، آهنگ 1x ... آهنگ 2x ... در اینجا کاربر دستور اجرای y که پلی‌لیست 5 آهنگ است را از آهنگ 4 ام می‌دهد. آهنگ 4y ... در اینجا کاربر دستور حذف آهنگ 2 و 3 و 1 پلی‌لیست x را می‌دهد (بر اساس ترتیب قبل از حذف آهنگ‌ها) یعنی برحسب ترتیب قبل از حذف آهنگ‌ها الان آهنگ 4,5,6 از x پخش نشده‌اند.

و حال دستور اجرای صف پخش می‌دهد.

بعد از اتمام صف پخش باید به اجرای پلی‌لیست y برگردد و آهنگ 5، آهنگ 1، آهنگ 2 و سپس آهنگ 3 پلی‌لیست y را به ترتیب پخش کند. و بعد به پلی‌لیست x برگردد و آهنگ‌های 4,5,6 را پخش کند.

این مدیریت پخش آهنگ‌ها (از پلی‌لیست‌ها و صف پخش) را باید با استفاده از stack انجام دهد.

لازم به ذکر است که اگر هنگام پخش پلی‌لیست X دستور پخش پلی‌لیست Y دستور پخش پلی‌لیست X مجدد داده شود باید دوباره تمام آهنگ‌های X شروع به پخش شوند (یعنی اجرای قدیمی اثری روی دستور پخش جدید ندارد) همچنین بعد از برگشت به پخش قبلی پلی‌لیست X، آهنگ‌های پخش شده در اجرای جدید تر تاثیری در پخش قدیمی ندارند و فقط مهم هست که آهنگ‌هایی از قبل پخش شده بودند (در اجرای قدیمی) دوباره اجرا نشوند. (اجرای جدید هم تاثیری روی دستور پخش قدیمی ندارد)