

3 課題: 簡単な音楽生成プログラム (2)

楽譜情報に基づき、前節の `note` 関数を利用して楽曲の波形データを生成するスクリプトを考える。

「きらきら星」の冒頭を生成する例を示す。この例では、「ドドソソララソー」の部分生成する。最初の「ド」から「ラ」までは、全て四分音符であるが、最後の「ソー」の部分は二分音符である。プログラムを簡単にするため、まず、四分音符は 1 秒、二分音符は 2 秒であるとする。

このスクリプトでは、`pitch` で、音の高さを指定する。指定方法は、音階を整数で表す。ここでは、中央の C を 60 とする。半音上の C# は 61、D は 62 となる。`duration` で、音の持続時間を秒数であらわす。つまり、`pitch(60)` と `duration(1)` で中央の C が 1 秒間続くことをあらわす。

```
% play music1
pitch = [60 60 67 67 69 69 67]
duration = [1 1 1 1 1 1 2]

fs = 8000;
n1 = 1;
xx = zeros(1,sum(duration)*fs + 1); % 領域の確保
for kk = 1:length(pitch)
    pp = pitch(kk);
    dd = duration(kk);
    tone = (1) % 前回作成した note 関数を用いる
    n2 = n1 + length(tone) - 1;
    xx(n1:n2) = xx(n1:n2) + tone;
    n1 = n2;
end
soundsc(xx, fs);
```

課題 2 (1) の部分を埋めてスクリプトを完成させよ。

上記のスクリプトでは、

```
pitch = [60 60 67 67 69 69 67]
duration = [1 1 1 1 1 1 2]
```

が、楽譜情報をあらわしている。ただ、このスクリプトのままでは、音符の長さが絶対時間であらわされているため、同じ楽譜でテンポが違う演奏を生成できない。そこで、`duration` は四分音符を 1 とした相対的な音符の長さをあらわすこととする。

課題 3 上記のスクリプトの楽譜情報の設定の次の部分で、テンポを BPM 値 (四分音符が 1 分間にいくつ演奏されるかを示す値) で設定したら、そのテンポで演奏を生成するようにスクリプトを修正せよ。

楽譜情報には休符もある。休符は、`pitch` が 0 で、必要な長さ分 (つまり、四分休符であれば、1) だけ `duration` がある音符だとして表現することにする。

課題 4 `note` 関数を休符にも対応するように修正し、休符を含む演奏を生成して、スクリプトの動作を確認せよ。