مسائل

۱. یک تابع با سرتیتر [S] = mySum(A) بنویسید که در آن A یک ماتریس یک بعدی است و S مجموع المان‌های A است. می‌توانید روش بازگشتی یا تکراری را برای حل مسئله به کار ببرید اما تابع آماده sum را استفاده نکنید.

حالات آزمون:

```
>> S = mySum([1 2 3])
S =
     6
>> S = mySum(1:100)
S =
  5050
```

۲. چندجمله‌ای چبیشف به صورت بازگشتی تعریف می‌شود. چندجمله‌ای چبیشف به دو نوع تقسیم می‌شود: اول و دوم. اولین نوع چندجمله‌ای چبیشف $T_n(x)$ دومین نوع آن $U_n(x)$ با رابطه بازگشتی زیر تعریف می‌شوند:

$$T_n(x) = \begin{cases} 1 & \text{if } n = 0 \\ x & \text{if } n = 1 \\ 2xT_{n-1}(x) - T_{n-2}(x) & \text{otherwise} \end{cases}$$

$$U_n(x) = \begin{cases} 1 & \text{if } n = 0 \\ 2x & \text{if } n = 1 \\ 2xU_{n-1}(x) - U_{n-2}(x) & \text{otherwise} \end{cases}$$

تابعی با سرتیتر [y] = myChebyshevPoly1(n,x) بنویسید که در آن Y، nامین چندجمله‌ای چبیشف نوع اول برحسب x باشد. مطمئن شوید که تابع می‌تواند مقدار ورودی را به صورت ماتریسی قبول کند. شما می‌توانید فرض کنید که x یک بردار سطری است. متغیر خروجی y نیز باید یک بردار سطری باشد.

حالات آزمون:

```
>> myChebyshevPoly1(0,1:5)
ans =
     1     1     1     1     1
>> myChebyshevPoly1(1,1:5)
ans =
     1     2     3     4     5
>> myChebyshevPoly1(3,1:5)
ans =
     1    26    99   244   485
```

سعی کنید چندجمله‌ای چبیشف را برای مقادیر مختلف x = 0:.01:5 رسم کنید.

۳. تابع آکرمن A یک تابع رشدکننده سریع است که با رابطه بازگشتی زیر تعریف می‌شود:

$$A(m,n) = \begin{cases} n+1 & \text{if } m = 0 \\ A(m-1,1) & \text{if } m > 0 \text{and } n = 1 \\ A(m-1, A(m, n-1)) & \text{if } m > 0 \text{and } n > 0 \end{cases}$$

تابعی با سرتیتر [A] = myAckermann(m,n) بنویسید که A تابع آکرمن محاسبه شده برای n و m باشد.

حالات آزمون:

```
>> A = myAckermann(1,1)
A =
     3
>> A = myAckermann(1,2)
A =
     4
>> A = myAckermann(2,3)
A =
     9
>> A = myAckermann(3,3)
A =
    61
>> A = myAckermann(3,4)
A =
   125
```

مقدار (۴-۴) myAckermann آنقدر بزرگ است که نوشتن آن مشکل است. اگر چه تابع آکرمن کاربر عملی بسیاری ندارد اما معکوس این تابع کاربردهای فراوانی در طراحی حرکت روباتیک دارد.

۴.m. تابع $C(n, k)$ که تعداد دفعات بدون تکرار k شی از n شی را محاسبه می‌کند، در بسیاری از کارهای کاربردی آماری استفاده می‌شود. برای مثال چه تعداد بستنی کرمی سه طعمی از میان ۱۰ طعم بستنی می‌توان داشت؟ برای حل این مسئله ما باید $C(10, 3)$ را محاسبه کنیم، تعداد راه‌های انتخاب سه طعم بدون تکرار از میان ۱۰ طعم مختلف. تابع C به صورت معمول k انتخاب از n نامیده می‌شود. می‌توانید فرض کنید که k و n اعداد صحیح 1x1 هستند.

اگر $n = k$ باشد، $C(n, k) = 1$ زیرا فقط یک راه برای انتخاب n شی از n شی وجود دارد. اگر $k = 1$ باشد، $C(n, k) = n$ است زیرا انتخاب هر n شی یک انتخاب از میان n شی است. برای سایر حالات $C(n, k) = C(n-1, k) + C(n-1, k-1)$ است. آیا می‌توانید ببینید چرا؟

تابعی با سرتیتر [N] = myNChooseK(n,k) بنویسید که تعداد انتخاب‌های k شی از میان n شی را بدون تکرار محاسبه کند.

حالات آزمون:

```
>> N = myNChooseK(10,1)
N =
        10
>> N = myNChooseK(10,10)
N =
         1
>> N = myNChooseK(10,3)
N =
       120
```

۵.m. در خریدهایی که با پول انجام می‌شود، فروشنده باید باقی پول اضافه پرداخت شده را پس بدهد. این معمولاً به نام پس دادن بقیه پول نامیده می‌شود. اسکناس‌ها و سکه‌های لازم که باید پس داده شود را می‌توان با رابطه بازگشتی تعریف کرد. اگر مبلغ پرداختی $100 بیش از بهای کالا

باشد، باقی باید یک اسکناس صد دلاری همراه با باقیمانده فراخوانی تابع برگشتی با 50$ منهای پول پرداخت شده باشد. اگر مبلغ پرداختی 100$ بیش از بهای کالا باشد، باقی باید یک اسکناس پنجاه دلاری همراه با باقیمانده فراخوانی تابع برگشتی با 50$ منهای پول پرداخت شده باشد. به همین منوال برای همه پول‌های رایج در امریکا می‌تواند بیان شود. پول‌های رایج در امریکا به دلار 100, 50, 20, 10,5, 1, 0.25, 0.10, 0.05 و 0.01 دلار هستند. برای این مسئله از ۲ دلاری که رایج نیست صرف نظر کرده‌ایم. شما می‌توانید فرض کنید cost و paid اعددی هستند که *paid* $>= cost$. متغیر خروجی change باید برداری ستونی مانند حالت نشان داده شده در حالات آزمون باشد.

با استفاده از تابع برگشتی تابعی با سرتیتر [change] = myChange(cost, paid) بنویسید که cost بهای کالا،paid مبلغ پرداخت شده و change ماتریسی حاوی لیست اسکناس‌ها و سکه‌هایی است که باید برگردانده شود. توجه: به حالت مبنا دقت کنید!

حالت آزمون:

```
>> format ban
>> C = myChange(27.57, 100)
C =
        50.00
        20.00
         1.00
         1.00
         0.25
         0.10
         0.05
         0.01
         0.01
         0.01
```

۶. نسبت طلائی، φ حد تابع $\frac{F(n+1)}{F(n)}$ است وقتی که n به سمت بینهایت میل می‌کند و F(n)، nامین عدد فیبوناچی است که می‌توان نشان دادکه دقیقاً $\frac{1+\sqrt{5}}{2}$ و یا تقریباً ۱.۶۲ است. می‌گوئیم $G(n) = \frac{F(n+1)}{F(n)}$ تقریب nام از نسبت طلائی است و $G(1) = 1$ است. همچنین می‌توان نشان داد که φهمچنین حد کسر نامتناهی زیر است:

$$\varphi = 1 + \cfrac{1}{1 + \cfrac{1}{1 + \cfrac{1}{1 + \ddots}}}.$$

تابع برگشتی با سرتیتر [G] = myGoldenRatio(n) بنویسید که در آن G، n امین تقریب نسبت طلائی بر حسب رابطه بازگشتی در کسر نامتناهی باشد. شما باید کسر نامتناهی را برای تقریب نسبت طلایی به کار ببرید و از تعریف G(n) استفاده نکنید. با این وجود برای هر دو حالت G(1) = 1 است.

حالت آزمون:

```
>> format long
>> G = myGoldenRatio(10)
G =
   1.618181818181818
>> (1 + sqrt(5))/2
ans =
   1.618033988749895
```

مطالعات نشان داده است که مستطیل با نسبت (یعنی طول تقسیم بـر عـرض) نزدیـک بـه ابعـاد طلایی بسیار زیباتر از سایر مستطیل‌هاست. نسبت تصویر در بسیاری از تلویزیـون‌هـای عـریض و پرده نمایش سینماها چقدر است؟

۷. بزرگترین مقسوم علیه مشترک دو عدد صحیح a و b بزرگترین عدد صحیحی است که اعداد بـر آن بخش پذیرند و تابعی که آن را محاسبه می‌کند با GCD(a,b) تعریف شده است. تـابع GCD را می‌توان به صورت بازگشتی نوشت. اگر b برابر 0 باشد آن‌گاه بزرگ‌ترین مقسوم علیـه مشـترک a است. در غیر این صورت GCD(a,b) = GCD(b, rem(a,b)) که rem(a,b) باقیمانده تقسیم a بر b است. فرض کنید a و b اعداد صحیح هستند. تـابع بازگشـتی بـا سـرتیتر [gcd] = myGCD(a,b) بنویسید که بزرگ‌ترین مقسوم علیه مشترک دو عدد a وb را به دست می‌آورد.

حالات آزمون:

```
>> gcd = myGCD(10,4)
gcd =
     2
>> gcd = myGCD(33,121)
gcd =
    11
>> gcd = myGCD(18,1)
gcd =
     1
```