IE 531 HW#1

TIANQI WU

1 (a) $\text{Rank}(A) = 3$

$\text{Rank}(A|y) = 3$

Since $\text{Rank}(A) = \text{Rank}(A|y)$

There is a solution $Ax = y$

(b) Pick 3 columns out of 5.

$\{a_1, a_2, a_3\} : S_1$

$\{a_1, a_3, a_5\} : S_2$

$\{a_2, a_3, a_4\} : S_3$

Results are:

$$\underset{S_1}{\begin{pmatrix} 1 \\ -7 \\ 1 \\ 0 \\ 0 \end{pmatrix}}, \quad \underset{S_2}{\begin{pmatrix} 8 \\ 0 \\ 1 \\ 0 \\ -7 \end{pmatrix}}, \quad \underset{S_3}{\begin{pmatrix} 0 \\ -7 \\ \frac{4}{3} \\ \frac{1}{3} \\ 0 \end{pmatrix}}$$

$$\lambda_1 S_1 + \lambda_2 S_2 + (1-(\lambda_1 + \lambda_2)) S_3$$

$$\begin{pmatrix} \lambda_1 + 8\lambda_2 \\ 7\lambda_2 - 7 \\ -\frac{1}{3}\lambda_1 - \frac{1}{3}\lambda_2 + \frac{4}{3} \\ -\frac{1}{3}\lambda_1 + \frac{1}{3}\lambda_2 + \frac{1}{3} \\ -7\lambda_2 \end{pmatrix}$$

1. (c) Output From Matlab:

```
>> IE531_hw1

A =

     2     0     0     6     2
     1    -1    -1     4     0
     2    -2     2     4     0
     2    -2     0     6     0
    -4     4    -8    -4     0
     4     0    -2    14     4


y =

     2
     7
    18
    16
   -40
     2


x =

 8 - 4*b - 7*a
   - 7*a - 7*b
         1 - b
            -b
 7*a + 7*b - 7

 A*x == y gives

ans =

     2 == 2
     7 == 7
   18 == 18
   16 == 16
 -40 == -40
     2 == 2

>>
```

1.(d) The solution (1) is verified by 1(c) already and the Matlab output of solution (2) is given below:

```
>> IE531_hw1

A =

     2     0     0     6     2
     1    -1    -1     4     0
     2    -2     2     4     0
     2    -2     0     6     0
    -4     4    -8    -4     0
     4     0    -2    14     4


y =

     2
     7
    18
    16
   -40
     2


x =

                        a
              - 7*a - 7*b
 11/3 - (7*b)/3 - (8*a)/3
  8/3 - (7*b)/3 - (8*a)/3
            7*a + 7*b - 7

A*x == y gives

ans =

     2 == 2
     7 == 7
    18 == 18
    16 == 16
   -40 == -40
     2 == 2
```

1.(e) Since the two solutions are linearly related. Knowing one indicates the other.

$a = -7\lambda 1 - 4\lambda 2 + 8$

$b = 8\lambda 1 + 5\lambda 2 - 8$

1.(d) After adding the two constraints, the rank of matrix of A is 5. Also, A has 5 columns. Hence, the matrix A has full column rank. It indicates that there will be no free variable and there can be only one solution to the combined set of equations.

2.(a)

$$f(x) = \begin{cases} x & 0 \le x \le 1 \\ 2-x & 1 < x \le 2 \\ 0 & 0 \text{ otherwise} \end{cases}$$

Let $u_i \in u$ i.i.d $[0,1]$

Find CDF $F(x)$, inverse transform $x_i = F^{-1}(u_i)$

Then, x_i is i.i.d r.v's

(b). $$F^{-1}(y) = \begin{cases} 1+\sqrt{2y}, & 0 \le y \le 0.5 \\ 1-\sqrt{2-2y}, & 0.5 < y \le 1 \end{cases}$$

$$F(x) = \begin{cases} \frac{1}{2}(x^2-2x+1), & 0 \le x \le 1 \\ \frac{1}{2}(-x^2+2x+1), & 1 < x \le 2. \\ 1 & \text{otherwise} \end{cases}$$

$$f(x) = \begin{cases} x-1 & 0 \le x \le 1 \\ -x+1 & 1 < x \le 2 \\ 0 & \text{otherwise} \end{cases}$$

(c) $$F^{-1}(y) = \begin{cases} 2-\sqrt{1-2y} & 0 \le y \le 0.5 \\ \sqrt{2y-1} & 0.5 < y < 1 \end{cases}$$

$$F(x) = \begin{cases} \frac{1}{2}(-x^2+4x-3) & 1 \le x \le 2 \\ \frac{1}{2}(x^2+1) & 0 \le x < 1 \\ 1 & \text{otherwise} \end{cases}$$

$$f(x) = \begin{cases} x & 0 \le x < 1 \\ -x+2 & 1 \le x \le 2 \\ 0 & \text{otherwise} \end{cases}$$

3. (a) The program (I) outputs $\boxed{\text{"kablooey"}}$ since it prints the current character and then iterates itself until the end.

The program (II) outputs $\boxed{\text{"yeoolbak"}}$ since it iterates itself until the end and then prints the string in reverse order.

(b). Given two number (m, n).

If $m < n$, the algorithm calls $gcd(m, n-m)$

If $m > n$, the algorithm calls $gcd(m-n, n)$

GCD of two numbers do not change if the smaller number is subtracted from the large number.

Hence, $gcd(m, n-m) = gcd(m,n)$ and $gcd(m-n, n) = gcd(m,n)$. After running the recursive function gcd, it will finally return n, when $n = m$. It works for any pair of numbers.

(c) Putchar(1234 % 10 + 'A') = putchar(4 + 'A')

E is printed, 1234 / 10 = 123

Putchar(123 % 10 + 'A') = Putchar(3 + 'A')

D is printed, 123 / 10 = 12

Putchar(12 % 10 + 'A') = Putchar(2 + 'A')

C is printed, 12 / 10 = 1

Putchar(1 % 10 + 'A') = putchar(1 + 'A')

B is printed.

Hence $\boxed{\text{EDCB}}$ is the output.

(d) The algorithm is not correct and will cause illegal moves. When $n=5$,

The top disk goes from A to D, <1>

A to B, <2>

A B C D

D to B <3>

A to B. <4>

The fourth move is illegal.

n-k disks stay in B for destination from the second step while the third step let k-1 disk move to B. It causes illegal move.