Revision questions based on computation

1. A square matrix has the same number of rows and columns and its size is defined by the variable **`n`**. The code fragment below performs the multiplication of two square matrices **`A`** and **`B`** and stores the result in matrix **`C`**.

```
int i,j,k,n;
// process rows
for (i = 0, i < n; i++)
  // process columns
  for (j = 0; j < n; j++)
    {
    c[i][j] = 0.0;
    // process row-column interactions and sum them into array c
    for (k = 0; k < n; k++)
      {
      c[i][j] = c[i][j] + a[i][k] * b[k][j];
      }
```

   What is the computational count ***in terms of `n`*** for the code. Do this for the case when
   i) you do **not** include the cost of the loop process
   ii) you do include the cost of the loop process

(i) For the case when you do **not** include the cost of the loop process the computational count for the code is

```
0      for (i = 0, i < n; i++)
             // process columns
0            for (j = 0; j < n; j++)
               {
n^2            c[i][j] = 0.0;
               // process row-column interactions and sum
                  them into array c
0              for (k = 0; k < n; k++)
                 {
3n^3             c[i][j] = c[i][j] + a[i][k] * b[k][j];
                 }
```

$= 3n^3 + n^2$

or $ass*(n^3+n^2) + add*n^3 + mul*n^3$

(ii) For the case when you do include the cost of the loop process the computational count is

```
1+2n for (i = 0, i < n; i++)
           // process columns
n(1+2n)         for (j = 0; j < n; j++)
             {
n^2          c[i][j] = 0.0;
             // process row-column interactions and sum
                them into array c
n^2(1+2n)       for (k = 0; k < n; k++)
               {
3n^3           c[i][j] = c[i][j] + a[i][k] * b[k][j];
               }
```

$= 1+2n + n(1+2n) + n^2 + n^2(1+2n) + 3n^3 = 5n^3 + 4n^2 + 3n +1$

or $ass*(1 + n + 2n^2 + n^3) + com*(n + n^2 + n^3) + add*(n + n^2 + 2n^3) + mul*(n^3)$

2. Show that a positive root exists for the function f(x)= x^3+x^2+x-1 . Use the secant method and Newton Raphson method to solve f(x)= x^3+x^2+x-1 for a positive root with a stopping criteria of $|x_i-x_{i-1}| < 0.001$

Secant solution

x0	f(x0)	x1	f(x1)	x2	num	denom	abs(x2-x1)
0	-1	1	2	0.33333333	2	3	0.666666667
1	2	0.33333333	-0.5185185	0.47058824	0.345679	-2.51852	0.137254902
0.33333333	-0.518518519	0.47058824	-0.2037452	0.55943001	-0.02797	0.314773	0.088841773
0.47058824	-0.203745166	0.55943001	0.04747224	0.54264169	0.004218	0.251217	0.016788319
0.55943001	0.04747224	0.54264169	-0.003112	0.54367454	5.22E-05	-0.05058	0.001032847
0.54264169	-0.003112035	0.54367454	-4.305E-05	**0.54368903**	-4.4E-08	0.003069	**1.44896E-05**
0.54367454	-4.30542E-05	0.54368903	3.9912E-08	0.54368901	5.78E-13	4.31E-05	1.34196E-08

Netwon-Raphson solution

xold	f(xold)	f'(xold)	xnew=xold-(f(xold)/f'(xold)	abs(xnew-xold)
1	2	6	0.666666667	0.333333333
0.666666667	0.407407	3.666667	0.555555556	0.111111111
0.555555556	0.035665	3.037037	0.543812105	0.011743451
0.543812105	0.000366	2.974819	**0.543689026**	**0.000123079**
0.543689026	3.99E-08	2.974171	0.543689013	1.34021E-08

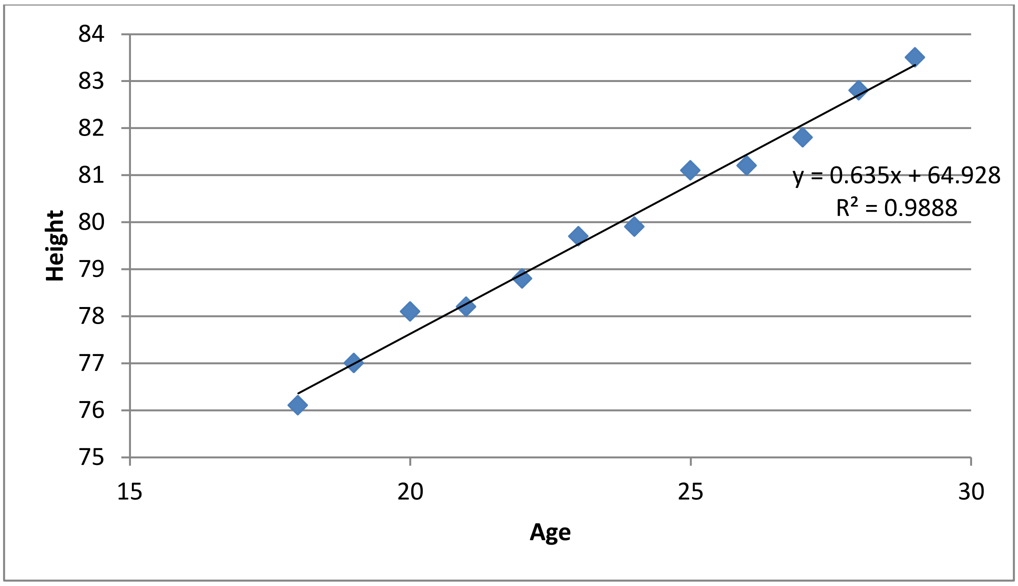
3. An Egyptian village was used as the site of a study of nutrition in developing countries. The data were obtained by measuring the heights (cm) of all 161 children in the village each month over several years. The data shows the mean heights for each age.

age	height (cm)
18	76.1
19	77
20	78.1
21	78.2
22	78.8
23	79.7
24	79.9
25	81.1
26	81.2
27	81.8
28	82.8
29	83.5

a. Obtain by hand, the equation of the least squares regression line.
b. Plot the regression line and use it to determine the average height for an average 27 year old.
c. Calculate the correlation coefficient for this data. What does it tell you about this data set?

a.

	age (x)	height (y)	y-mean y	x-mean x	(x-mean x)(y-mean y)	(x-mean x)^2
	18	76.1	-3.75	-5.5	20.625	30.25
	19	77	-2.85	-4.5	12.825	20.25
	20	78.1	-1.75	-3.5	6.125	12.25
	21	78.2	-1.65	-2.5	4.125	6.25
	22	78.8	-1.05	-1.5	1.575	2.25
	23	79.7	-0.15	-0.5	0.075	0.25
	24	79.9	0.05	0.5	0.025	0.25
	25	81.1	1.25	1.5	1.875	2.25
	26	81.2	1.35	2.5	3.375	6.25
	27	81.8	1.95	3.5	6.825	12.25
	28	82.8	2.95	4.5	13.275	20.25
	29	83.5	3.65	5.5	20.075	30.25
n=12						
sum	282	958.2			90.8	143
mean	23.5	79.85				
			b1= 0.634965		b0=	64.92832168



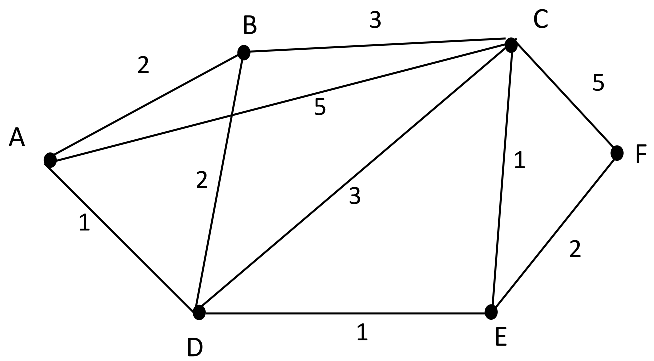
b.

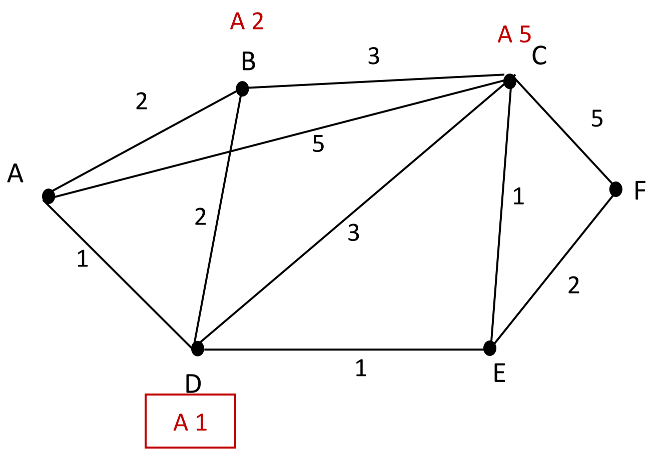
y = 0.635x + 64.928 when x=27 y=82.073 so average height of 27 yr old is 82.1cms.

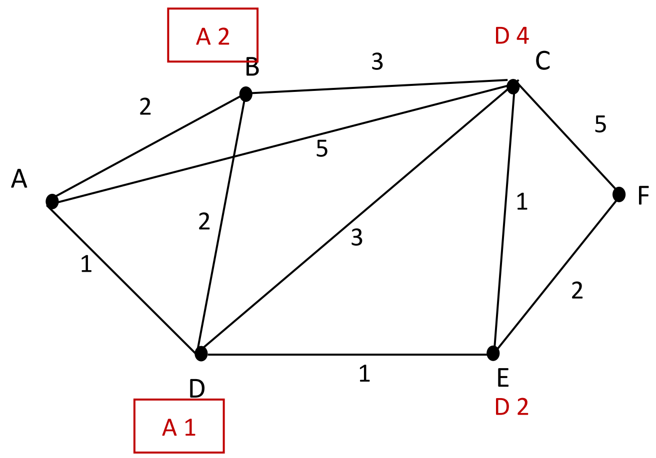
c.

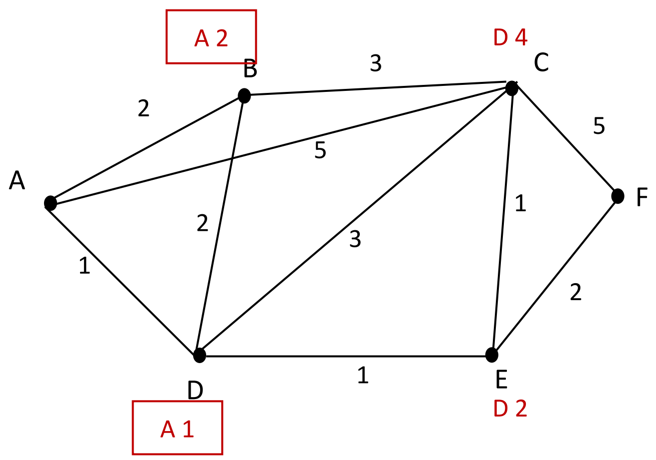
R^2 = 0.9888 -? very high degree of correlation so data could be said to follow linear trend. But is it sensible for extrapolation? We do not grow in height every year of our lives?

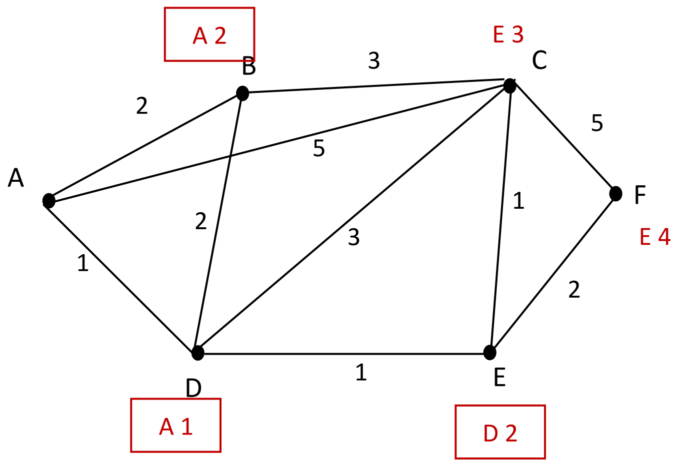
4. Find the shortest path from A to F. Show all your working clearly identifying the path and the cost incurred.

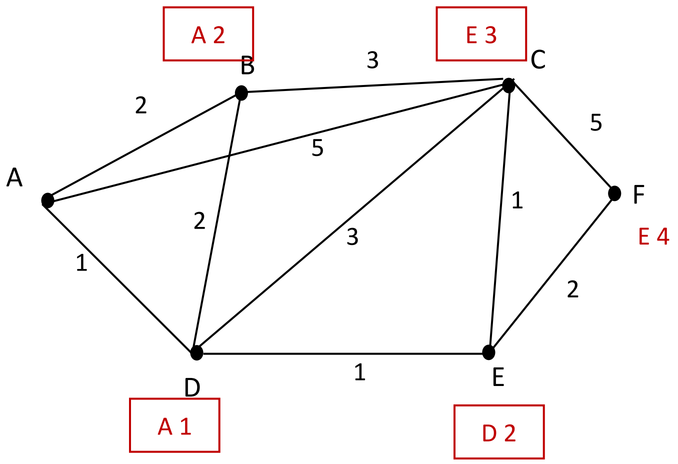


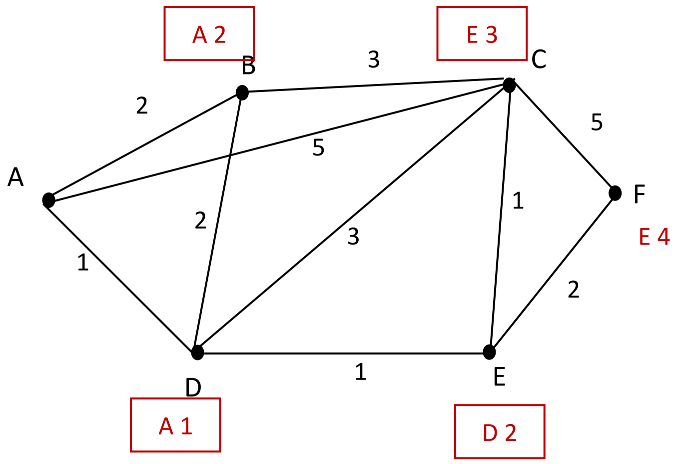


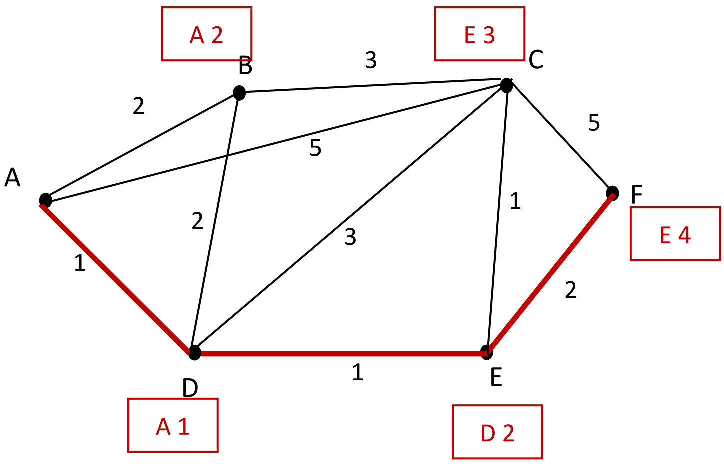












Min route from A to F is ADEF with a cost=4

5. Use Simpsons Rule with 6 strips (n=6) to calculate by hand the area under the curve

$$f(x) = 10x^2 - 6x^3 - 90x^4 + 400x^5$$

from a lower limit a=1 to an upper limit b=3 .

Work out the analytical (exact answer) and then calculate the absolute relative true error for this numerical scheme.

Simpsons rule

2n= 6.000000000

i	xi	f(xi)	4*f(xi)	2*f(xi)		
0.000000000	1.000000000	314.000000000				
1.000000000	1.333333333	1404.707818930	5618.831275720			
2.000000000	1.666666667	4449.588477366		8899.176954733		
3.000000000	2.000000000	11352.000000000	45408.000000000		solution=	44156.872427984
4.000000000	2.333333333	24976.288065844		49952.576131687	abs true err=	12.872427984
5.000000000	2.666666667	49345.316872428	197381.267489712		abs rel true err=	0.000291601
6.000000000	3.000000000	89838.000000000				

6. Given a hash table of size n = 10 and two hash functions $h1$ and $h2$:

$h1$(x) = (sum of the values of first and last letters of x) modulus of n

$h2$(x) = ((value of the last letter of x) modulus of (n-1)) + 1

where the value of a letter is determined by its position in the alphabet (e.g. value(a)=1, value(b)=2, etc)

Here are some pre-computed hash values as an example:

x	$h1$(x)	$h2$(x)
Pugh	4	9
Lai	1	1
Margo	8	7
Annie	6	6
Gonzales	6	2
Pam	9	5
Cherry	8	8
Lenon	6	6
Kiri	0	1
Barbara	3	2

For **each** hash function draw the resulting hash table after inserting, in order, the following words:

Barbara, Pam, Gonzales, Lai, Pugh, Annie, Kiri, Cherry, Lenon, Margo.

(i) Use linear probing when collision occurs

(ii) Using chaining when collision occur

Use linear probing if collision occurs.

Using $h1$ hash function

Index	Word
0	Kiri
1	Lai
2	Lenon
3	Barbara
4	Pugh
5	Margo
6	Gonzales
7	Annie
8	Cherry
9	Pam

Using $h2$ hash function

Index	Word
0	Margo
1	lai
2	barbara
3	gonzales
4	kiri
5	pam
6	annie
7	lenon
8	Cherry
9	Pugh

(iii) Using chaining instead when collision occurs, draw the hash tables again for the words provided above for each hash function.

Using *h*1 hash function

Index	Word		
0	Kiri		
1	Lai		
2			
3	Barbara		
4	Pugh		
5			
6	Gonzales	Annie	Lenon
7			
8	Cherry	Margo	
9	Pam		
10			
11			
12			

Using *h*2 hash function

Index	Word	
0		
1	Lai	
2	Barbara	Gonzales
3		
4		
5	Pam	
6	Annie	Lenon
7	Margo	
8	Cherry	
9	Pugh	