Question 1b:

```
> restart:
  poly_A := expand((x+1)*(2*x^4-3*x^3+5*x^2+3*x-1)):
  poly_B := expand((x+1)*(7*x^4+5*x^3-12*x^2-x+4)):
  print(poly_A);
  print(poly_B);
```

$$2x^5 - x^4 + 2x^3 + 8x^2 + 2x - 1$$

$$7x^5 + 12x^4 - 7x^3 - 13x^2 + 3x + 4 \quad (1)$$

```
> (* Extended Euclidean Algorithm for Q[x] implementation *)


  e_gcd := proc(a::polynom, b::polynom)
      local A, B, r, s, t, k, s_f, t_f, g:

      A, B, k := a, b, 0:
      r[k], r[k+1] := A, B:
      s[k], s[k+1] := 1, 0:
      t[k], t[k+1] := 0, 1:

     while r[k+1] <> 0 do
          r[k+2] := rem(r[k], r[k+1], x, 'q'):
         s[k+2] := expand(s[k]-q*s[k+1]):
         t[k+2] := expand(t[k]-q*t[k+1]):

        printf("\n\nIteration: %d\nRemainder: %a\nS%a: %a\nT%a:
  %a\n", k+2, r[k+2], k+2, s[k+2], k+2, t[k+2]):
          k := k+1:
      od:

      g := r[k]:
      s_f := s[k]:
      t_f := t[k]:
     printf("\n\nWe stop as the remainder is 0 and the GCD is the
  last non-zero remainder i.e. the computed GCD is: %a.
  \n\nThe value of S and T are follows:\nS: %a\nT: %a.\n\n",
  g/lcoeff(g), s_f/lcoeff(g), t_f/lcoeff(g));

     if (expand((s_f/lcoeff(g)*A) + (t_f/lcoeff(g)*B))) = g/lcoeff
  (g) then
         printf("Output satisfies SA + TB = G.\n");
      else
         printf("Output does not satisfy SA + TB = G.\n");
       fi:
```

```
    end proc:
> e_gcd(poly_A,poly_B);


Iteration: 2
Remainder: -31/7*x^4+4*x^3+82/7*x^2+8/7*x-15/7
S2: 1
T2: -2/7


Iteration: 3
Remainder: 26971/961*x^3+35819/961*x^2+4172/961*x-4676/961
S3: 49/31*x+3976/961
T3: -14/31*x-175/961


Iteration: 4
Remainder: -75208821/103919263*x^2-119863608/103919263*
x-44654787/103919263
S4: 961/3853*x^2+10010737/103919263*x-6779855/14845609
T4: -1922/26971*x^2+13537607/103919263*x-23028443/103919263


Iteration: 5
Remainder: -9248814407/21096478639*x-9248814407/21096478639
S5: 727434841/75208821*x^3+2287470817156/1961972513427*x^2
-33593044876643/1961972513427*x+17349217038587/1961972513427
T5: -207838526/75208821*x^3+3784012123619/653990837809*x^2
-6793202222310/653990837809*x+4122269324684/1961972513427


Iteration: 6
Remainder: 0
S6: -21096478639/1321259201*x^4-105482393195/9248814407*
x^3+253157743668/9248814407*x^2+21096478639/9248814407*
x-84385914556/9248814407
T6: 42192957278/9248814407*x^4-63289435917/9248814407*
x^3+105482393195/9248814407*x^2+63289435917/9248814407*
x-21096478639/9248814407


We stop as the remainder is 0 and the GCD is the last non-zero
remainder i.e. the computed GCD is: x+1.

The value of S and T are follows:
S: -182609/8277*x^3-22012/8277*x^2+323261/8277*x-166949/8277
T: 52174/8277*x^3-36413/2759*x^2+65370/2759*x-39668/8277.
```

Output satisfies SA + TB = G.

Confirming answer using Maples command:

```
> gcdex(poly_A,poly_B,x,'s','t');
```

$$x+1 \tag{2}$$

```
> s;
```

$$-\frac{182609}{8277}x^3-\frac{22012}{8277}x^2+\frac{323261}{8277}x-\frac{166949}{8277} \tag{3}$$

```
> t;
```

$$\frac{52174}{8277}x^3-\frac{36413}{2759}x^2+\frac{65370}{2759}x-\frac{39668}{8277} \tag{4}$$

Question 1c:

```
> a_x := x^3 + 1:
  b_x := x^2 + 1:
  c_x := x^2:

  e_gcd(a_x, b_x):
```

Iteration: 2
Remainder: -x+1
S2: 1
T2: -x

Iteration: 3
Remainder: 2
S3: x+1
T3: -x^2-x+1

Iteration: 4
Remainder: 0
S4: 1/2*x^2+1/2
T4: -1/2*x^3-1/2

We stop as the remainder is 0 and the GCD is the last non-zero remainder i.e. the computed GCD is: 1.

```
The value of S and T are follows:
S: 1/2*x+1/2
T: -1/2*x^2-1/2*x+1/2.

Output satisfies SA + TB = G.
```

```
> s_x := (x+1)/2:
  t_x := (-x^2-x+1)/2:

  e_x := expand(a_x * s_x + b_x * t_x):

  s_p := c_x * s_x:
  t_p := c_x * t_x:

  q_x := quo(s_p, b_x, x, 'r'):
  sigma := r:

  tau := t_p + q_x * a_x:

  res := expand(sigma * a_x + tau * b_x):

  printf("\nValue of Sigma: %a.\nValue of Tau: %a.\nValue of Sigma
  * a_x + Tau * b_x: %a = c_x.", sigma, tau, res);
```

```
Value of Sigma: -1/2*x-1/2.
Value of Tau: (x^3+1)*(1/2*x+1/2)+x^2*(-1/2*x^2-1/2*x+1/2).
Value of Sigma * a_x + Tau * b_x: x^2 = c_x.
```