```
/**********************************
 * EECS2031 - Assignment 1
 * Filename: poly.c
 * Author: Wang, Yang
 * Email: infi5959512@hotmail.com
 * Login ID: infi999
 **********************************/

#include "poly.h"


/*
 Initialize all coefficients and exponents of the polynomial to zero.
 */
void init_polynom( int coeff[ ], int exp[ ] )
{

    /* ADD YOUR CODE HERE */
    int i;
    for (i=0;i<ASIZE;i++){
        coeff[i] = 0;
        exp[i] = 0;
    }

}  /* end init_polynom */


/*
 Get inputs from user using scanf() and store them in the polynomial.
 */
void get_polynom( int coeff[ ], int exp[ ] )
{

    /* ADD YOUR CODE HERE */
    int input[ASIZE];
    //scan the first element to get the length of coeff and exp
    scanf("%d", &input[0]);

    int j;
    //scan the rest of input
    for(j=1; j<=(input[0]*2);j++){
        scanf("%d", &input[j]);
    }
    int index;
    int indexI = 0;
    // use index and indexI because we dont need put first input into coeff[] and exp[]
    for(index=1; index<j-1;index++,indexI++){
        coeff[indexI] = input[index];
        index++;
        exp[indexI] = input[index];
    }


}  /* end get_polynom */


/*
 Convert the polynomial to a string s.
 */
void polynom_to_string( int coeff[ ], int exp[ ], char s[ ] )
{

    /* ADD YOUR CODE HERE */
```

```
	int index = 0;
	int indexI;

	for (indexI = 0;indexI < ASIZE ;indexI++){
		// polynomial has only signle 0
		if (coeff[0] == 0){
		   s[index++] = '0';
		   break;
		}
		else if (coeff[indexI] != 0){
		   int coe = coeff[indexI];
		   int temp = coeff[indexI];
		   int zero = -1;
		   // zero counter (for case which coe has more than one zero)
		    while ((temp % 10) == 0){
		       temp = coe % 10;
		       coe = coe / 10;
		       zero++;}
		   coe = coeff[indexI];
		   // set the sign for coeff
		   if (coeff[indexI] < 0){
		       s[index++] = '-';
		       coe = coeff[indexI] * -1;}
		   else if (coeff[indexI] > 0 && index > 0){
		       s[index++] = '+';}

		   int a = 0;
		   int b = 0;

		   if (coe > 1){
		   if (coe >= 1){
		   // reverse the coe
		   while(coe >= 1){
		      a = coe % 10;
		      coe = coe / 10;
		      b = (b*10) + a;}

		   coe = b;
		   a = 0;
		   // convert int to char and save coeff[] into s[]
		   while(coe >= 1){
		      a = coe % 10;
		      coe = coe / 10;
		      s[index++] = a + '0';}
		   // add 0 if coe has more than one 0
		   int z1 = 0;
		   while(z1<zero){
		      s[index++] = '0';
		      z1++;}}}



		   int exp1 = exp[indexI];
		   // special case input 1 1 0
		   if (exp[indexI] == 0 && coeff[indexI] == 1){
		       s[index++] = coeff[indexI] + '0';}
		   if (exp[indexI] == 0 && coeff[indexI] == -1){
		       s[index++] = 1 + '0';}
		   // put x^- into s[]
		   if (exp[indexI] != 0){
		      s[index++] = 'x';
		   if (exp[indexI] != 1 && exp[indexI] != -1){
		      s[index++] = '^';}
		   if (exp[indexI] < 0){
		      s[index++] = '-';
```

```
                exp1 = exp[indexI] * -1;}



            int a = 0;
            int b = 0;
            int zero = -1;
            int temp = exp[indexI];
            while((temp%10) == 0){
                 temp = exp1%10;
                 exp1 = exp1/10;
                 zero++;}
            exp1 = exp[indexI];
            if (exp1 > 1){
            // reverse exp1
            while(exp1 >= 1){
               a = exp1 % 10;
               exp1 = exp1 / 10;
               b = (b*10) + a;}

            exp1 = b;
            a = 0;
             // convert int to char and save exp[] into s[]
            while(exp1 >= 1){
               a = exp1 % 10;
               exp1 = exp1 / 10;
               s[index++] = a + '0';}
            int z1 = 0;
            // add 0 if exp1 has more than one 0
            while(z1<zero){
            s[index++] = '0';
            z1++;
            }}



}}}
// set s[] as String
s[index] = '\0';

}  /* end polynom_to_string */



/*
 Evaluate the polynomial for the value of x and store the result p(x) in variable result.
 */
void eval_polynom( int coeff[ ], int exp[ ], double x, double *result )
{

    /* ADD YOUR CODE HERE */
  int i, j;
  double sum = 0;
  double x1;

      for (i=0; i<ASIZE; i++){
           x1 = x;

        if (exp[i] != 0){

          for (j=0; j<exp[i]-1;j++){

           x1 = x1 * x;}

           x1 = x1 * coeff[i];
```

```
            sum = sum + x1;}

          //exp = 0
          else{
             x1 = 1 * coeff[i];
             sum = sum + x1;}}
              *result = sum;

}  /* end eval_polynom */



/*
 Add two polynomials and the result is stored in the first polynomial (arrays co1[] and ex1[]).
 */
void add_polynom( int co1[ ], int ex1[ ], int co2[ ], int ex2[ ] )
{

    /* ADD YOUR CODE HERE */

          int coTemp[ASIZE], exTemp[ASIZE];
          int indexT;
          int indexI1 = 0;
          int indexI2 = 0;

          for (indexT = 0; indexT<ASIZE; indexT++){
               if (ex1[indexI1] == ex2[indexI2]){
                 coTemp[indexT] = co1[indexI1] + co2[indexI2];
                 exTemp[indexT] = ex1[indexI1];
                 indexI1++; indexI2++;}
               else if (ex1[indexI1] > ex2[indexI2]){
                 coTemp[indexT] = co1[indexI1];
                 exTemp[indexT] = ex1[indexI1++];}
               else if (ex1[indexI1] < ex2[indexI2]){
                 coTemp[indexT] = co2[indexI2];
                 exTemp[indexT] = ex2[indexI2++];}}

           // copy result to co1[] and ex1[]
           int indexI3;
           for (indexI3 = 0; indexI3<ASIZE; indexI3++){
                 co1[indexI3] = coTemp[indexI3];
                 ex1[indexI3] = exTemp[indexI3];}


}  /* end add_ polynom */


/************************* END OF FILE ***************************/
```