

```

12. if (coefficient)
13. { attach(coefficient, terms[startA].expon); }
14. startA++; startB++;
15. break;

16. case 1: /* a expon > b expon
17. attach(terms[startA].coef, terms[startA].expon);
18. startA++;
19. } /* add in remaining terms of A(x) */
20. for(; startA < finishA; startA++)
21. attach(terms[startA].coef, terms[startA].expon);
/* add in remaining terms of B(x) */
22. for(; startB < finishB; startB++)
23. attach(terms[startB].coef, terms[startB].expon);
* finishD = avail - 1; 24.

```

```

} 25.

1. void attach(float coefficient, int exponent)
2. { /* add a new term to the polynomial */
3. if (avail == MAX_TERMS) {
4. printf(stderr, "Too many terms in the polynomial\n");
5. exit(EXIT_FAILURE);
6. }
7. terms[avail].coef = coefficient;
8. terms[avail].expon = exponent;
9. }

```

Proof of Correctness

• Its easy to see that the attach function is correct.

Assumptions: 1. Each polynomial is represented as a sequence of terms in descending order of exponents. That is, if the polynomial has terms stored from terms[startA] to terms[finishA], then: terms[startA].expon > terms[startA+1].expon > ... > terms[finishA].expon