```
#include <iostream>
#include <fstream>
#include <cstdlib>

using namespace std;

/*
 * -> k, x[100]
 * <- pares[100][2],npares
 * em que pares é um vector de pares tais que x[i]+x[j]=k
 * e npares dá-nos o números de pares encontrados.
 */
int obtempares(int k,int x[],int pares[][2]) {
  int npares=0,i,j;

  i=0;
  while (i <= 99) {
    j=i;
    while (j <= 99) {
      if ((x[i]+x[j])==k) {
        pares[npares][0] = x[i];
        pares[npares][1] = x[j];
        npares++;
      }
      j++;
    }
    i++;
  }
  return(npares);
}

/*
 * resolução do exercício 34
 * -> k, ficheiro contendo 100 inteiros
 * <- ficheiro contendo pares de inteiros
 *
 * é de notar que este programa não tem nenhuma interacção com o
 * utilizador, além dos argumentos dados na linha de comando. Todas as
 * entradas e saídas são feitas através de ficheiros.
*/
int main(int argc, char *argv[]){
  int i,xi,k;
  ifstream de(argv[2]); // declara um canal de entrada
  ofstream para(argv[3]); // declara um canal de saída
  int x[100]; //vector de 100 inteiros
  int pares[4950][2]; // vector de 4950 pares de inteiros
  //(4950 = Combinaçlsões de 100 2 a 2)
  int npares; // número de pares encontrados

  if (argc < 4)
    cout << "\nUtilização: ./exer35 <n> <ficheiro entrada> <ficheiro de saída>\n\n";
  else {
    k = atoi(argv[1]);
    if (!de) cout << "Não foi possível abrir o ficheiro de entrada";
    else {
      if (!para) cout << "Não foi possível abrir o ficheiro de saída";
      else {
        //começa por ler os inteiros para um vector
        i=0;
        while (de >> xi) {
          x[i] = xi;
          i++;
        }
        // determina os pares de elementos de x tais que xi+xj=k
        npares = obtempares(k,x,pares);
        // escreve o vector de pares de inteiros num ficheiro
        i = 0;
        while (i < npares) {
          para << "(" << pares[i][0] << "," << pares[i][1] << ");";
```



```
          i++;
        }
      }
    }
  }
  return 0;
}
```