Write a C++ program to convert between unsigned integer number representations. For the internal representation we can use an array of int to store the coefficients, where the the first entry is the base and the last entry is $-1$. Thus, to represent the number 43 decimal in binary we would use the array

$$(2, 1, 1, 0, 1, 0, 1, -1).$$

Write C++ functions of the form

```cpp
int most_significant(int x, int base);
void tobase(int x, int base, int *&rep);
void convert_base(int *inrep, int base, int *&outrep);
```

The function `most_significant` calculates the size of the array of coefficients for the representation. For the above example (binary representation of 43) the value returned is 5 (the coefficient of $2^5$ is the last entry of the array). The function `tobase` takes an integer and stores the representation for $x$ in `rep`, `tobase` should allocate the memory for `rep`. Explain the reason for the data type of the third parameter of `tobase`. The function `convert_base` is equivalent to `tobase`, except it takes an arbitrary representation for the integer to convert.

Use the program to convert 157 decimal to binary, and then from binary to ternary (base 3) representation.