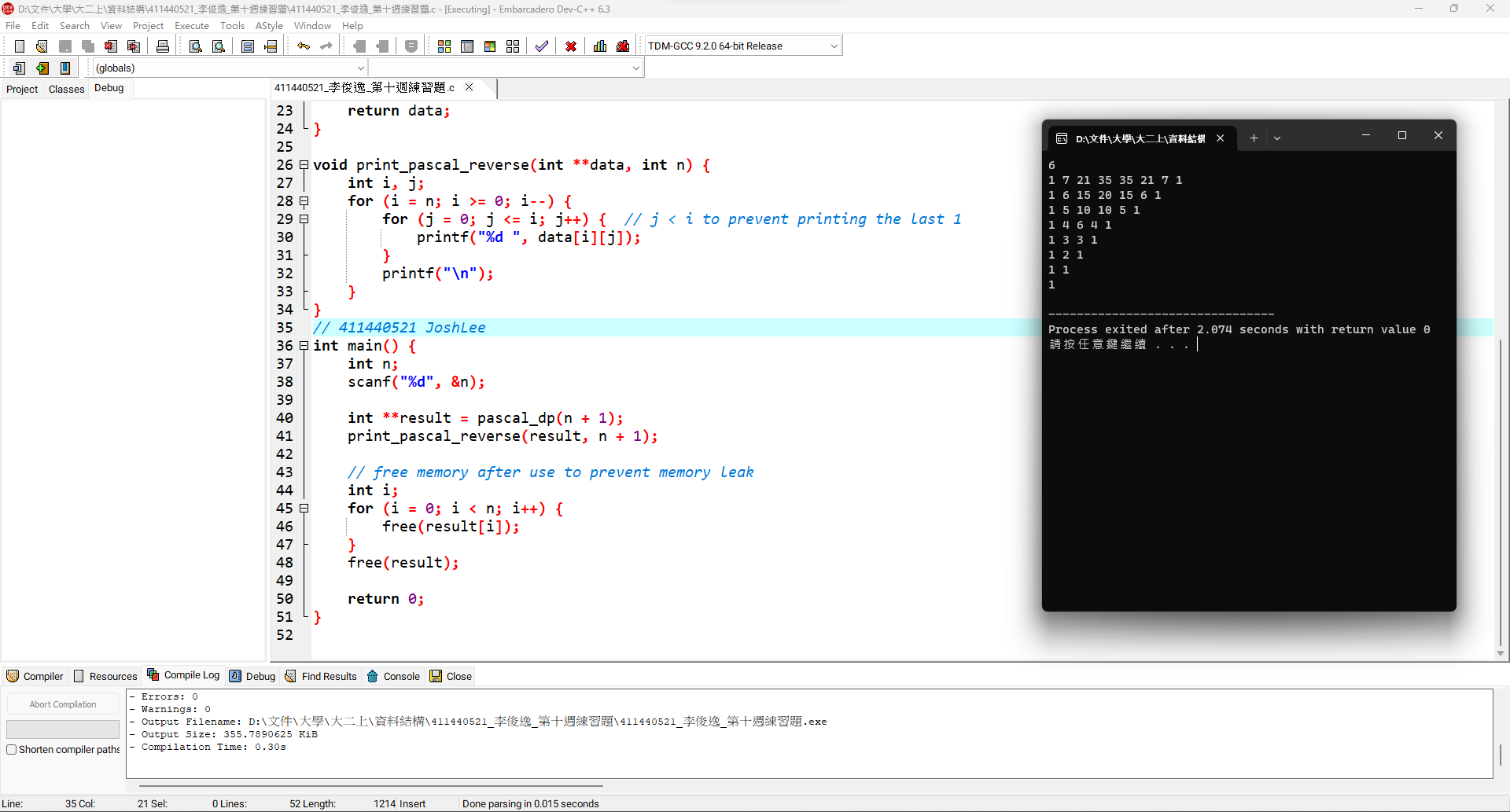


1

2

程式碼:

|  |
| --- |
| */\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  *\* Author: 411440521 JoshLee \**  *\* Date: 2023/11/2 \**  *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/*  #include <stdio.h>  #include <stdlib.h>  int \*\*pascal\_dp(int n) {  int col, row;  int \*\*data = (int \*\*)malloc(sizeof(int \*) \* (n + 1)); *// store row data*  for (col = 0; col <= n; col++) {  data[col] = (int \*)malloc(sizeof(int) \* (col + 1)); *// store row value*  data[col][0] = 1;  for (row = 1; row <= col; row++) {  data[col][row] = data[col - 1][row - 1] + data[col - 1][row];  }  data[col][col] = 1;  }  return data;  }  void print\_pascal\_reverse(int \*\*data, int n) {  int i, j;  for (i = n; i >= 0; i--) {  for (j = 0; j <= i; j++) { *// j < i to prevent printing the last 1*  printf("%d ", data[i][j]);  }  printf("\n");  }  }  *// 411440521 JoshLee*  int main() {  int n;  scanf("%d", &n);  int \*\*result = pascal\_dp(n + 1);  print\_pascal\_reverse(result, n + 1);  *// free memory after use to prevent memory leak*  int i;  for (i = 0; i < n; i++) {  free(result[i]);  }  free(result);  return 0;  } |