|  |  |
| --- | --- |
| **EX06-06.cpp:** *Dealing with Bitwise Operators* | |
| **Line#** | **Code** |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34 | #include <iostream>  using namespace std;  static void showBits(int bits32) {  for (int i = 0; i < 32; i++) {  cout << ((bits32 & 0x80000000) != 0);  bits32 <<= 1;  }  cout << endl;  }  int main() {  int x = 3, y = 5;  cout << "x:\t"; showBits(x);  cout << "y:\t"; showBits(y);  cout << "x&y:\t"; showBits(x & y); //Bitwise AND  //cout<<"x|y:\t"; showBits(x|y); //Bitwise OR  //cout<<"x^y:\t"; showBits(x^y); //Bitwise XOR  //cout<<"~x:\t"; showBits(~x); //Inversion  //cout<<"x>>1:\t"; showBits(x>>1); //Shift Right  //cout<<"x>>2:\t"; showBits(x>>2); //Shift Right  //cout<<"x<<1:\t"; showBits(x<<1); //Shift Left  //cout<<"x<<2:\t"; showBits(x<<2); //Shift Left  //x = -1333333333;  //cout<<"x:\t"; showBits(x);  //cout<<"x>>1:\t"; showBits(x>>1); //Shift Right  //cout<<"x<<1:\t"; showBits(x<<1); //Shift Left  return 0;  } |

|  |  |
| --- | --- |
| EX06-06b.cpp | |
| Line# | Code |
| **1**  **2**  **3**  **4**  **5**  **6**  **7**  **8**  **9**  **10**  **11**  **12**  **13**  **14**  **15**  **16**  **17**  **18**  **19**  **20**  **21**  **22**  **23**  **24**  **25**  **26**  **27**  **28**  **29**  **30**  **31**  **32**  **33**  **34**  **35**  **36**  **37**  **38**  **39**  **40**  **41**  **42**  **43**  **44**  **45**  **46**  **47**  **48**  **49**  **50**  **51**  **52**  **53**  **54**  **55**  **56**  **57**  **58** | #include <iostream>  **using namespace** std;  #define HAS\_CASH 0x00000001  #define HAS\_CAR 0x00000002  #define HAS\_CERT 0x00000004  #define HAS\_CAREER 0x00000008  #define HAS\_CONDO 0x00000010  **static void** showBits(**int** bits32) {  **for** (**int** i = 0; i < 32; i++) {  cout << ((bits32 & 0x80000000) != 0);  bits32 <<= 1;  }  cout << endl;  }  **int** main() {  /\*  cout << "HAS\_CASH:\t"; showBits(HAS\_CASH);  cout << "HAS\_CAR:\t"; showBits(HAS\_CAR);  cout << "HAS\_CERT:\t"; showBits(HAS\_CERT);  cout << "HAS\_CAREER:\t"; showBits(HAS\_CAREER);  cout << "HAS\_CONDO:\t"; showBits(HAS\_CONDO);  \*/  /\*  int status = HAS\_CASH | HAS\_CAR;  cout << "status:\t"; showBits(status);  //I sold my car  status &= ~HAS\_CAR;  cout << "status:\t"; showBits(status);  //I found a job  status |= HAS\_CAREER;  cout << "status:\t"; showBits(status);  //Got a new car and also new certification  status |= (HAS\_CAR | HAS\_CERT);  cout << "status:\t"; showBits(status);  //showBits(status);  //showBits(HAS\_CAR);  int mask = HAS\_CAR | HAS\_CERT;  if ((status & mask) != 0) cout << "Has CAR OR Cert" << endl;  if ((status & mask) == mask) cout << "Has CAR AND Cert" << endl;  //int mask = HAS\_CAR | HAS\_CERT;  //cout << "mask:\t"; showBits(mask);  //status |= mask; //SET Operation  //status ^= mask; //TOGGLE Operation  //cout << "~mask:\t"; showBits(~mask);  //status &= ~mask; //CLEAR Operation  //cout << "status:\t"; showBits(status);  \*/  **return** 0;  } |