**Белорусский государственный технологический университет**

**Факультет информационных технологий**

**Кафедра программной инженерии**

**Программа “Конвертер систем счисления”**

По дисциплине «Основы алгоритмизации и программирования»

Выполнил:

Студент 1 курса 10 группы ПИ

Макаревич Кирилл Витальевич

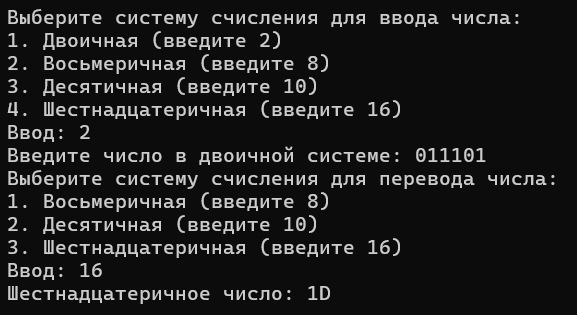
Проверил: Белодед Николай Иванович

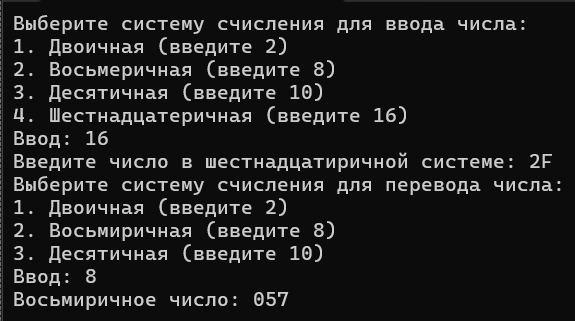
2024, Минск

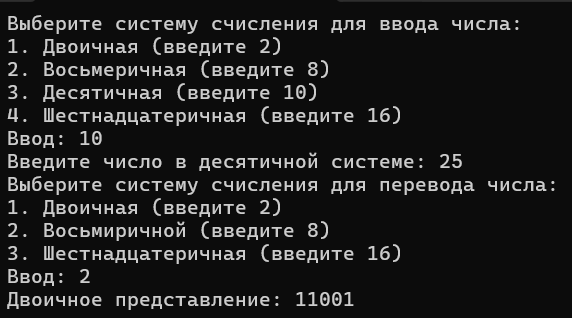
Код программы:

|  |
| --- |
| #include <stdio.h>  #include <Windows.h>  #include<iostream>  #include<string>  using namespace std;  //функция для перевода из двоичной в восьмиричную  string binaryToOctal(string binary) {  // тут я преобразую двоичное число к длине, кратной 3, добавив нули слева  int length = binary.length();  int remainder = length % 3;  if (remainder != 0) {  int padding = 3 - remainder; // тут я смотрю сколько нулей нужно добавить  for (int i = 0; i < padding; i++) {  binary = '0' + binary;  }  }  string octal;  for (int i = 0; i < binary.length(); i += 3) {  string binaryGroup = binary.substr(i, 3); // метод substr берёь 3 символа начиная с i элемента строки  char octalDigit;  // тут я написал таблицу преобразования триад двоичных цифр в восьмеричные цифры  if (binaryGroup == "000") octalDigit = '0';  else if (binaryGroup == "001") octalDigit = '1';  else if (binaryGroup == "010") octalDigit = '2';  else if (binaryGroup == "011") octalDigit = '3';  else if (binaryGroup == "100") octalDigit = '4';  else if (binaryGroup == "101") octalDigit = '5';  else if (binaryGroup == "110") octalDigit = '6';  else if (binaryGroup == "111") octalDigit = '7';  else octalDigit = '0';  octal += octalDigit;  }  return octal;  }  //функция для перевода из двоичной в десятичную  int binaryToDecimal(string binary) {  int decimal = 0;  int base = 1; // тут начальная степень двойки: 2^0 = 1  for (int i = binary.length() - 1; i >= 0; i--) {  if (binary[i] == '1') {  decimal += base;  }  base \*= 2;  }  return decimal;  }  //функция для перевода из двоичной в шестнадцатиричную  string binaryToHex(string binary) {  // тут я преобразую двоичное число к длине, кратной 3, добавив нули слева  int length = binary.length();  int remainder = length % 4;  if (remainder != 0) {  int padding = 4 - remainder; // тут я смотрю сколько нулей нужно добавить  for (int i = 0; i < padding; i++) {  binary = '0' + binary;  }  }  string hexadecimal;  // Проходим по двоичной строке группами по 4 цифры  for (int i = 0; i < binary.length(); i += 4) {  string binaryGroup = binary.substr(i, 4); // метод substr берёт 4 символа начиная с i элемента строки  char hexDigit;  // таблица преобразования тетрад двоичных цифр в шестнадцатеричные цифры  if (binaryGroup == "0000") hexDigit = '0';  else if (binaryGroup == "0001") hexDigit = '1';  else if (binaryGroup == "0010") hexDigit = '2';  else if (binaryGroup == "0011") hexDigit = '3';  else if (binaryGroup == "0100") hexDigit = '4';  else if (binaryGroup == "0101") hexDigit = '5';  else if (binaryGroup == "0110") hexDigit = '6';  else if (binaryGroup == "0111") hexDigit = '7';  else if (binaryGroup == "1000") hexDigit = '8';  else if (binaryGroup == "1001") hexDigit = '9';  else if (binaryGroup == "1010") hexDigit = 'A';  else if (binaryGroup == "1011") hexDigit = 'B';  else if (binaryGroup == "1100") hexDigit = 'C';  else if (binaryGroup == "1101") hexDigit = 'D';  else if (binaryGroup == "1110") hexDigit = 'E';  else if (binaryGroup == "1111") hexDigit = 'F';  else hexDigit = '0';  hexadecimal += hexDigit;  }  return hexadecimal;  }  //функция для перевода из восьмиричной в двоичную  string octalToBinary(string octal) {  string binary;  for (int i = 0; i < octal.length(); i++) {  int digit = octal[i] - '0'; //преобразуем символ индекса i в восьмиричной системе в его целочисленное представление , смотрим код символа в таблице кодировки windows-1251  if (digit < 0 || digit > 7) {  cout << "Некорректная восьмеричная цифра: " << octal[i] << endl;  return "";  }  // смотрим триады цифр на соответсвие их в восьмиричной  if (digit == 0) binary += "000";  else if (digit == 1) binary += "001";  else if (digit == 2) binary += "010";  else if (digit == 3) binary += "011";  else if (digit == 4) binary += "100";  else if (digit == 5) binary += "101";  else if (digit == 6) binary += "110";  else if (digit == 7) binary += "111";  }  if (binary.empty()) return "0";// если строка будет пустая, то возвращаем символ "0"  return binary;  }  //функция для перевода из восьмиричной в десятичную  int octalToDecimal(string octal) {  string binary = octalToBinary(octal);  int bindec = binaryToDecimal(binary);  return bindec;  }  //функция для перевода из восьмиричной в шестнадцатиричную  string octalTohex(string octal)  {  string binary= octalToBinary(octal);  string binhex = binaryToHex(binary);  return binhex;  }  //функция для перевода из десятичной в двоичную  string decimalToBinary(string decimalStr) {  int decimal = stoi(decimalStr); // stoi преобразует строку в целое число  if (decimal == 0) return "0";  string binary;  while (decimal > 0) {  int remainder = decimal % 2; // остаток от деления на 2  binary += to\_string(remainder); // тут мы преобразуем remainder из типа int в string и добавляем остаток к строке  decimal /= 2;  }  reverse(binary.begin(), binary.end()); //reverse переворачивает строку, чтобы получить правильный порядок в ответе  return binary;  }  //функция для перевода из десятичной в восьмиричную  string decimalToOctal(string decimalStr)  {  string binary = decimalToBinary(decimalStr);  string binoct = binaryToOctal(binary);  return binoct;  }  //функция для перевода из десятичной в шестнадцатиричную  string decimaltoHex(string decimalStr)  {  string binary = decimalToBinary(decimalStr);  string binhex = binaryToHex(binary);  return binhex;  }  //функция для перевода из шестнадцатиричной в двоичную  string hexToBinary(string hex) {  string binary;  for (int i = 0; i < hex.length(); i++) {  char c = hex[i];  if (c >= 'a' && c <= 'f') {  c -= 32; // тут мы превращаем строчные буквы в прописные (мы занем , что в кодировке windows-1251 строчные буквы больше прописных на 32(в 10с.с.)  }  if (c == '0') binary += "0000";  else if (c == '1') binary += "0001";  else if (c == '2') binary += "0010";  else if (c == '3') binary += "0011";  else if (c == '4') binary += "0100";  else if (c == '5') binary += "0101";  else if (c == '6') binary += "0110";  else if (c == '7') binary += "0111";  else if (c == '8') binary += "1000";  else if (c == '9') binary += "1001";  else if (c == 'A') binary += "1010";  else if (c == 'B') binary += "1011";  else if (c == 'C') binary += "1100";  else if (c == 'D') binary += "1101";  else if (c == 'E') binary += "1110";  else if (c == 'F') binary += "1111";  else {  cout << "Некорректная шестнадцатеричная цифра: " << c << endl;  return "";  }  }  if (binary.empty()) { //empty проверяет пустая ли данная строка  return "0";  }  else {  return binary;  }  }  //фунция для перевода из шестнадцатиричной в восьмиричную  string hexToOctal(string hex) {  string binary = hexToBinary(hex);  string binoct = binaryToOctal(binary);  return binoct;  }  //фунция для перевода из шестнадцатиричной в десятичную  int hexToDec(string hex) {  string binary = hexToBinary(hex);  int bindec = binaryToDecimal(binary);  return bindec;  }  int main() {  SetConsoleCP(1251);  SetConsoleOutputCP(1251);  int inputnum, reinputnum;  string number;  cout << "Выберите систему счисления для ввода числа:\n";  cout << "1. Двоичная (введите 2)\n";  cout << "2. Восьмеричная (введите 8)\n";  cout << "3. Десятичная (введите 10)\n";  cout << "4. Шестнадцатеричная (введите 16)\n";  cout << "Ввод: ";  cin >> inputnum;  switch (inputnum) {  case 2: {  cout << "Введите число в двоичной системе: ";  cin >> number;  cout << "Выберите систему счисления для перевода числа:\n";  cout << "1. Восьмеричная (введите 8)\n";  cout << "2. Десятичная (введите 10)\n";  cout << "3. Шестнадцатеричная (введите 16)\n";  cout << "Ввод: ";  cin >> reinputnum;  switch (reinputnum) {  case 8: {  string octal = binaryToOctal(number);  cout << "Восьмеричное число: " << octal << endl;  break;  }  case 10:  {  int decimal = binaryToDecimal(number);  cout << "Десятичное число: " << decimal << endl;  break;  }  case 16:  {  string hexadecimal = binaryToHex(number);  cout << "Шестнадцатеричное число: " << hexadecimal << endl;  break;  }  default:  cout << "Такого варианта нет\n";  return 1;  }  break;  }  case 8: {  cout << "Введите число в восьмиричной системе: ";  cin >> number;  cout << "Выберите систему счисления для перевода числа:\n";  cout << "1. Двоичная (введите 2)\n";  cout << "2. Десятичная (введите 10)\n";  cout << "3. Шестнадцатеричная (введите 16)\n";  cout << "Ввод: ";  cin >> reinputnum;  switch (reinputnum)  {  case 2:  {  string binary = octalToBinary(number);  cout << "Восьмиричное число: " << binary << endl;  break;  }  case 10: {  int binary = octalToDecimal(number);  cout << "Десятичное число: " << binary << endl;  break;  }  case 16: {  string binary = octalTohex(number);  cout << "Десятичное число: " << binary << endl;  break;  }  default:  {  cout << "Такого варианта нет\n";  return 1;  break;  }  }  break;  }  case 10: {  cout << "Введите число в десятичной системе: ";  cin >> number;  cout << "Выберите систему счисления для перевода числа:\n";  cout << "1. Двоичная (введите 2)\n";  cout << "2. Восьмиричной (введите 8)\n";  cout << "3. Шестнадцатеричная (введите 16)\n";  cout << "Ввод: ";  cin >> reinputnum;  switch (reinputnum)  {  case 2: {  string binary = decimalToBinary(number);  cout << "Двоичное представление: " << binary << endl;  break;  }  case 8:  {  string octal = decimalToOctal(number);  cout << "Восьмеричное представление: " << octal << endl;  break;  }  case 16:  {  string hex = decimaltoHex(number);  cout << "Шестнадцатиричное представление: " << hex << endl;  }  default:  {  cout << "Такого варианта нет\n";  return 1;  break;  }  }  break;  }  case 16:{  cout << "Введите число в шестнадцатиричной системе: ";  cin >> number;  cout << "Выберите систему счисления для перевода числа:\n";  cout << "1. Двоичная (введите 2)\n";  cout << "2. Восьмиричная (введите 8)\n";  cout << "3. Десятичная (введите 10)\n";  cout << "Ввод: ";  cin >> reinputnum;  switch (reinputnum)  {  case 2:  {  string binary = hexToBinary(number);  cout << "Двоичное число: " << binary << endl;  break;  }  case 8:  {  string binary = hexToOctal(number);  cout << "Восьмиричное число: " << binary << endl;  break;  }  case 10:  {  int binary = hexToDec(number);  cout << "Десятичное число: " << binary << endl;  break;  }  default:  {  cout << "Неверный выбор системы счисления.\n";  return 1;  }  }  }  }  return 0;  } |

Скринтшоты результата:







Блок-схема алгоритма