Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

"Белгородский государственный технологический университет им. В. Г. Шухова"

(БГТУ им. В.Г. Шухова)

Институт информационных технологий и управляющих систем

Кафедра программного обеспечения вычислительной техники

и автоматизированных систем

**Лабораторная работа № 1**

**по дисциплине математическая логика и теория алгоритмов**

**тема: Логика высказываний**

**Выполнил: студент группы ПВ-223**

**Игнатьев Артур Олегович**

**Проверил: старший преподаватель**

**Куценко Дмитрий Александрович**

Белгород 2023

Лабораторная работа № 1

**Тема:** Логика высказываний

**Цель работы:** Разработать программный модуль, способный находить значение формы (представленной в нормальной форме) на данной интерпретации. Разработать программу, способную считывать формулу логики высказываний в одной из нормальных форм (по выбору пользователя) и находить значения данной формулы на вводимых пользователем интерпретации.

**Содержание отсчёта**

1. Название и цель лабораторной работы.

2. Решение предложенных в теоретической части задач.

3. Программа на выбранном языке программирования в виде исходных кодов (с поясняющими комментариями) и в электронном варианте для демонстрации на ЭВМ.

4. Спецификация программы с указанием основных структур данных и алгоритмов.

5. Наборы текстовых данных.

**Вариант 3**

Теоретическое задание:

2.6. Если в огороде нет бузины, то в Киеве нет дядьки.

7.3. Постройте таблицу истинности, соответствующую следующей формуле:

8.18. Используя таблицу истинности, доказать равность формулы:

10.5. Используя равносильные преобразования доказаь, выполняется ли следующее соотношение:

12.2. Найти ДНФ для формулы:

14.6. Упростить вид формулы, используя равносильные преображения:

19.4. Проверить эквивалентность следующей формулы, преобразуя формулу с обеих сторон от знака “≡” к одной и той же нормальной форме:

23.5. Приведением к совершенной нормальной форме доказать неравносильность следующей формулы:

30.4. Для следующего выражения найти двойственные:

35.10. Выяснить является ли первая формула логическим следствием остальных:

38.6. Доказать правильность умозаключения:

49.2. Найти все следствия из посылки

Практическое задание:

Разработать программу, решающую задачи согласно своему варианту. Программа должна считывать формулу логики высказываний в указанной нормальной форме. Алгоритмы, выполняющие решения задачи, должны содержатся в отдельном модуле.

Форма КНФ:

1. Программа должна строить таблицу истинности введённой формулы.

4. Программа должна отыскивать все интерпретации, на которых введённая формула принимает истинное значение.

Решение заданий:

Теоретическая часть:

2.6. Если в огороде нет бузины, то в Киеве нет дядьки.

P: "в огороде нет бузины"

Q: "в Киеве нет дядьки"

P → Q

Если условие P истинно, то следствие Q тоже должно быть истинно. В противном случае, если условие P ложно, то утверждение P → Q не накладывает никаких ограничений на Q.

7.3. Постройте таблицу истинности, соответствующую следующей формуле:

X→(X˅Y)

|  |  |  |  |
| --- | --- | --- | --- |
| X | Y | X˅Y | X→(X˅Y) |
| 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 |

Как видно из таблицы, для всех комбинаций значений X и Y, выражение X → (X ∨ Y) имеет значение 1 (истина).

8.18. Используя таблицу истинности, доказать равность формулы:

X&0≡0

|  |  |  |  |
| --- | --- | --- | --- |
| X | X&0 | 0 | X&0≡0 |
| 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 |

Из таблицы видно, что для всех значений X (как 0, так и 1), результат операции X & 0 равен 0. Таким образом, формула X & 0 ≡ 0 верна для всех значений X.

10.5. Используя равносильные преобразования доказать, выполняется ли следующее соотношение:

Доказательства равносильности данного соотношения:

Доказали равносильность данного соотношения:

12.2. Найти ДНФ для формулы:

Нахождение дизъюнктивной нормальной формы (ДНФ):

Найдена дизъюнктивная нормальная формф (ДНФ):

14.6. Упростить вид формулы, используя равносильные преображения:

Найдём упрощенный вид данной формулы:

Упрощенная формула равна просто Y.

19.4. Проверить эквивалентность следующей формулы, преобразуя формулу с обеих сторон от знака “≡” к одной и той же нормальной форме:

Начнём с левой стороны:

Перейдем к правой стороне:

(Q ∨ P ∨ (Q ∧ P))

Обе стороны формулы упростились до 1. Исходная формула и преобразованная формула эквивалентны:

23.5. Приведением к совершенной нормальной форме доказать неравносильность следующей формулы:

Начнем с левой стороны:

Преобразуем правую сторону:

Совершенные нормальные формы (СНФ) для левой и правой стороны формулы:

Левая сторона:

Правая сторона:

Сравнив обе СНФ они не эквивалентны.

30.4. Для следующего выражения найти двойственные:

Двойственным выражением для является .

35.10. Выяснить является ли первая формула логическим следствием остальных:

Для того чтобы выяснить, является ли первая формула "X̅" логическим следствием остальных формул, используем таблицу истинности.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | Y | Z | X̅ | X ≡ Y |  | (X ≡ Y) & |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 |

Когда формула "X̅" принимает значение 1, то "X̅" равно 1 только в строках, где X равно 0. Таким образом, в каждой из этих строк формула "X̅" является истиной.

Формулы не являются одновременно истинными. Следовательно, любая формула является их следствием.

Таким образом, заданное логическое следствие верно.

38.6. Доказать правильность умозаключения:

Для того чтобы выяснить, является ли первая формула "" логическим следствием остальных формул, используем таблицу истинности.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C |  |  |  |  |  |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |

Когда формула "" принимает значение 1, то "" равно 1 только в строках, где A равно 0. Таким образом, в каждой из этих строк формула "" является истиной.

Формулы не являются одновременно истинными. Следовательно, любая формула является их следствием.

Таким образом, заданное логическое следствие верно.

49.2. Найти все следствия из посылки

Составим конъюнкцию посылок и эквивалентными преобразованиями приведём её к СКНФ:

Логические следствия:

1. – посылка 1
2. – посылка 2

Ответ: логическими следствиями указанных посылок являются следующие формуды: , , .

Практическая часть:

Файл lab1.h

#include <stdio.h>  
#include <string.h>  
#include <windows.h>  
  
// Функция определения значения переменной в заданной интерпретации  
bool evaluate(char variable, bool interpretation[]);  
  
// Функция вычисления значения формулы для заданной интерпретации  
bool evaluateFormula(char formula[], bool interpretation[]);  
  
// Функция для построения таблицы истинности  
void truthTable(char formula[], int numVariables);  
  
// Функция для поиска всех интерпретаций, при которых формула истинна  
void findTrueInterpretations(char formula[], int numVariables);

Файл lab1.c

#include "lab1.h"  
  
bool evaluate(char variable, bool interpretation[]) {  
 int index = variable - 'A'; // Индекс переменной в массиве интерпретации  
 return interpretation[index];  
}  
  
bool evaluateFormula(char formula[], bool interpretation[]) {  
 int len = strlen(formula);  
 for (int i = 0; i < len; i++) {  
 if (formula[i] == '&') {  
 continue; // Пропускаем конъюнкцию  
 } else if (formula[i] == '|') {  
 continue; // Пропускаем дизъюнкцию  
 } else if (formula[i] == '!') {  
 i++; // Переходим к следующему символу после '!'  
 bool result = !evaluate(formula[i], interpretation); // Отрицание  
 return result;  
 } else {  
 return evaluate(formula[i], interpretation); // Переменная  
 }  
 }  
 return false; // В случае ошибки  
}  
  
void truthTable(char formula[], int numVariables) {  
 int numRows = 1 << numVariables; // 2 в степени numVariables  
 printf("Таблица истинности:\n");  
  
 // Заголовок таблицы  
 for (int i = 0; i < numVariables; i++) {  
 printf("%c\t", 'A' + i);  
 }  
 printf("Формула\n");  
  
 // Заполнение таблицы  
 for (int row = 0; row < numRows; row++) {  
 bool interpretation[numVariables];  
  
 // Заполняем интерпретацию текущей строки  
 for (int i = 0; i < numVariables; i++) {  
 interpretation[i] = (row & (1 << i)) != 0; // Установка битов в соответствии с текущей строкой  
 printf("%d\t", interpretation[i]);  
 }  
  
 // Вычисляем значение формулы для текущей интерпретации  
 bool result = evaluateFormula(formula, interpretation);  
 printf("%d\n", result);  
 }  
}  
  
void findTrueInterpretations(char formula[], int numVariables) {  
 int numRows = 1 << numVariables; // 2 в степени numVariables  
 printf("Истинные интерпретации:\n");  
  
 for (int row = 0; row < numRows; row++) {  
 bool interpretation[numVariables];  
  
 // Заполняем интерпретацию текущей строки  
 for (int i = 0; i < numVariables; i++) {  
 interpretation[i] = (row & (1 << i)) != 0; // Установка битов в соответствии с текущей строкой  
 }  
  
 // Вычисляем значение формулы для текущей интерпретации  
 bool result = evaluateFormula(formula, interpretation);  
  
 // Если значение формулы истинно, выводим интерпретацию  
 if (result) {  
 for (int i = 0; i < numVariables; i++) {  
 printf("%d ", interpretation[i]);  
 }  
 printf("\n");  
 }  
 }  
}

Файл main.c

int main() {  
 // Устанавливаем кодировку вывода в UTF-8 для корректного отображения  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int numVariables;  
 printf("Введите количество переменных: ");  
 scanf("%d", &numVariables);  
  
 char formula[100];  
 printf("Введите формулу в КНФ: ");  
 scanf("%s", formula);  
  
 // Построение таблицы истинности  
 truthTable(formula, numVariables);  
  
 // Поиск интерпретаций, при которых формула истинна  
 findTrueInterpretations(formula, numVariables);  
 return 0;  
}

Вывод: на этой лабораторной работе я разработал программный модуль, способный находить значение формы (представленной в нормальной форме) на данной интерпретации. Разработал программу, способную считывать формулу логики высказываний в одной из нормальных форм (по выбору пользователя) и находить значения данной формулы на вводимых пользователем интерпретации.