**ТЕМА ЗА ПРОЕКТ**

**Интерпретатор на Prolog**

1. Описание на проекта

Този проект представлява Prolog интерпретатор, написан на Haskell. Целта на интерпретатора е да зарежда Prolog база данни от файл, да обработва заявки от потребителя и да връща решения според логическите факти и правила, дефинирани в базата данни.

1. Основни функционалности
2. **Зареждане на база данни:**

* Интерпретаторът може да чете Prolog база данни от текстов файл.
* Базата данни може да съдържа факти и правила, написани в ограничен Prolog синтаксис.

1. **Обработка на заявки:**

* Потребителят може да въвежда заявки чрез командния ред.
* Интерпретаторът намира решения за въведената заявка според заредената база данни.

1. **Работа с множество решения:**

* Интерпретаторът поддържа множество решения за дадена заявка (ако съществуват).

1. Структура на проекта. Основни модули и типове
2. **Типове:**
   * Fact: Представя факт в Prolog базата данни (напр. parent(john, mary)).
   * Rule: Представя правило с глава и тяло (напр. ancestor(X, Y) :- parent(X, Y).).
   * Atom: Представя предикат с неговите аргументи.
   * Substitution: Представя единична замяна (например X = john).
   * Context: Списък от факти и правила, които съставляват базата данни.
3. **Парсери:**
   * Функции за парсване на факти, правила и цели Prolog файлове.
4. **Унификация:**
   * Реализация на алгоритъма за унификация, който сравнява два атома и връща необходимите замени, за да ги направи равни.
5. **Решаване на заявки:**
   * Функции за търсене на съответствие между заявка и факт или правило в базата данни.