Sprawozdanie PSIR

Projekt

Jakub Sulikowski 318839
Michał Filipczyk 318761
Kacper Samuś 318829

Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych

6 lutego 2024

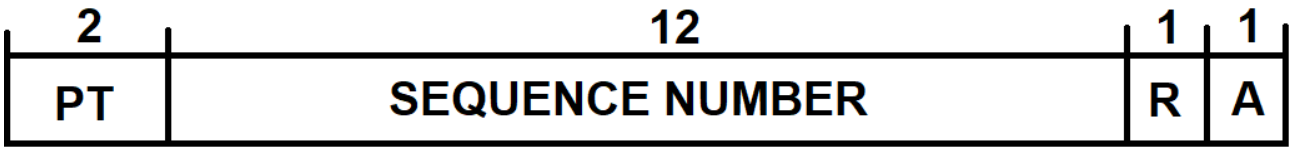
Spis treści

1. Zadanie projektowe

Celem projektu było stworzenie warstwy programistycznej pośredniej między systemem operacyjnym, a aplikacjami, w postaci przestrzeni krotek. Ponadto, stworzenie protokołu ALP oraz dwóch aplikacji testujących.

2. Specyfikacja ALP

ALP jest to binarny protokół warstwy aplikacji działający na UDP, używany do komunikacji klienta z serwerem operującym przestrzenią krotek.



Rys. 1: Nagłówek pakietu protokołu ALP

Typ	ilość bitów
PAYLOAD TYPE	2
SEQUENCE NUMBER	12
RDP RESULT	1
ACKNOWLEDGE	1

2.1. Payload type

Zawiera informacje o typie przesyłanej wiadomości i wartości jakie przyjmuje odpowiednio:
— ack: potwierdź otrzymanie krotki o zadanych parametrach - wartość: 0b00
— out: umieść krotkę o zadanych parametrach w przestrzeni krotek - wartość: 0b01
— rdp: pobierz daną krotkę z przestrzeni krotek i zwróć RDP_RESULT = 1 jeżeli istnieje w przeciwnym razie zwróć RDP_RESULT = 0 - operacja nie blokująca - wartość: 0b10
— inp: pobierz a następnie wyjmij krotkę z przestrzeni krotek - operacja nie blokująca - wartość: 0b11

2.2. Sequence number (Nie zaimplementowano)

Numer sekwencji służy do identyfikowania czy nie utracono żadnego pakietu oraz czy pakiet należy do tego samego hosta w sesji.

2.3. RDP_RESULT

Jest wynikiem operacji rdp().

2.4. Acknowledge

Ta flaga ma wartość jeden gdy wysyłane jest potwierdzenie otrzymania wiadomości.

3. Opis implementacji serwera

3.1. Opis implementacji ALP na serwerze

Najważniejszą cechą naszej implementacji jest wprowadzenie "rutyn" odbierania i wysyłania. Rutyna wysyłania zawiera wysłanie oryginalnej wiadomości oraz oczekiwanie na potwierdzenie przez określony czas, po którym jeżeli nie otrzyma potwierdzenia następuje ponowne przesłanie oryginalnej wiadomości. Rutyna wysyłania jest

funkcją blokującą. W rutynie odbierania otrzymywana jest wiadomość oraz wysyłane jest potwierdzenie otrzymania wiadomości. Rutyna odbierania jest funkcją nieblokującą. Dodatkowo zaimplementowano mechanizm sprawdzania błędów protokołu.

3.2. Opis implementacji API przestrzeni krotek

API przestrzeni krotek jest swoistym łącznikiem między protkołem aplikacji, a przestrzenią krotek. W celu ułatwienia komunikacji zaimplementowano strukturę tuple_t odpowiedzialną za przechowywanie danych krotki - jej nazwę, liczbę pól oraz pola.

3.3. Opis implementacji przestrzeni krotek na serwerze

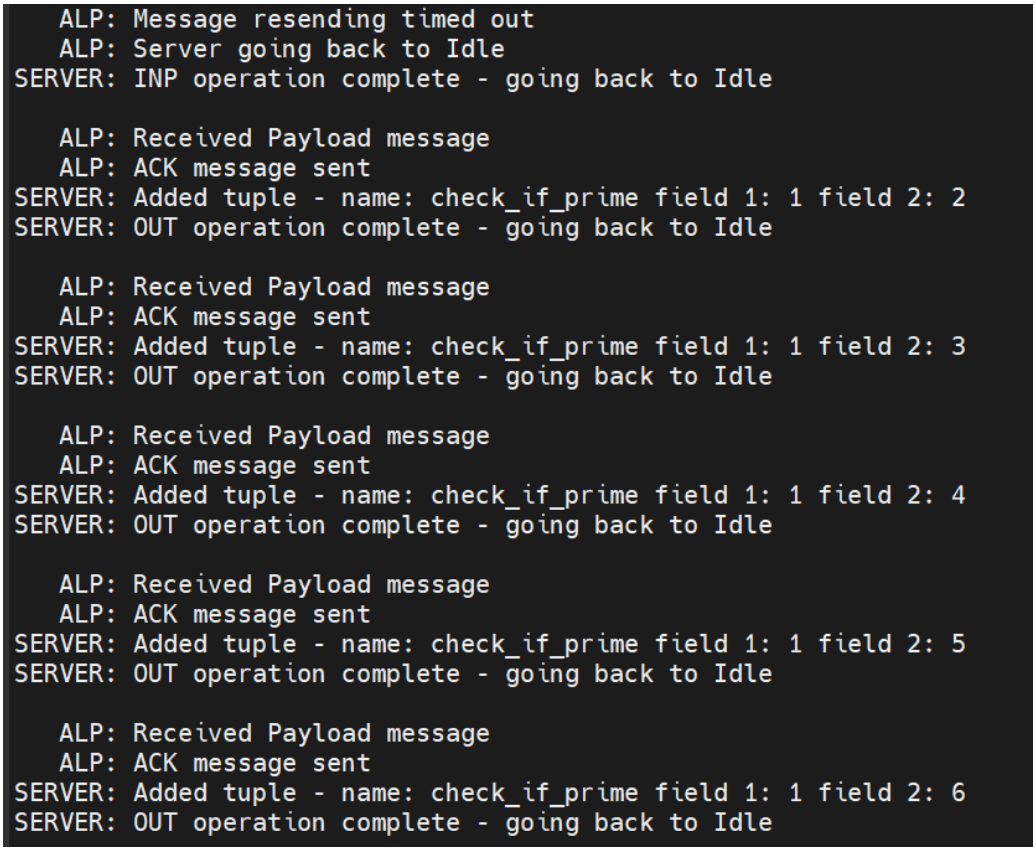
Przestrzeń krotek zaimplementowano jako linked_list co ułatwia operacje na przestrzeni. Na przestrzeni krotek można wykonać 3 operacje:
— Dodać krotkę na końcu listy
— Zwrócić krotkę i usunąć na podstawie indeksu
— Zwrócić krotkę bez usuwania na podstawie indeksu

```
   ALP: Received Payload message
   ALP: ACK message sent
SERVER: Removed tuple from Tuple Space
   ALP: Payload message sent
   ALP: Recived ACK message
SERVER: INP operation complete - going back to Idle

   ALP: Received Payload message
   ALP: ACK message sent
SERVER: Removed tuple from Tuple Space
   ALP: Payload message sent
   ALP: Recived ACK message
SERVER: INP operation complete - going back to Idle
```

Rys. 2: Udane wykonanie operacji INP

```
   ALP: Received Payload message
   ALP: ACK message sent
SERVER: Given tuple does not exist in tuple space
   ALP: Payload message sent
   ALP: Recived ACK message
SERVER: INP operation complete - going back to Idle

   ALP: Received Payload message
   ALP: ACK message sent
SERVER: Given tuple does not exist in tuple space
   ALP: Payload message sent
   ALP: Recived ACK message
SERVER: INP operation complete - going back to Idle
```

Rys. 3: Nieudane wykonanie operacji INP - krotka nie istnieje w przestrzeni - wtedy serwer zwraca pustą krotkę



Rys. 4: Wyświetlane powiadomienia po stronie serwera

Mechanizm porównywania danej krotki z danym szablonem jest zrealizowany rekurencyjnie.

4. Opis implementacji węzłów

4.1. Opis implementacji ALP na serwerze

Podobnie jak w implementacji serwera działanie protokołu opiera się na rutynach. Jedyną różńicą między implementacją serwerową, a węzłową jest to, że w węzłach nie zaimplementowano mechanizmu sprawdzania błędów, który figuruje w implementacji protokołu na serwerze.

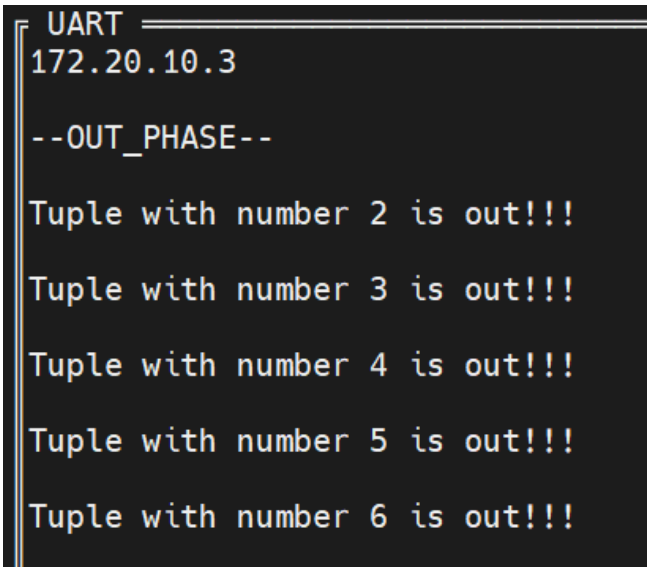
4.2. Opis implementacji API przestrzeni krotek

W węzłach zaimplementowano odpowiednie funkcje reprezentujące operacje na przestrzeni krotek. Podobnie jak w implementacji na serwerze skorzystano ze struktury tuple_t.

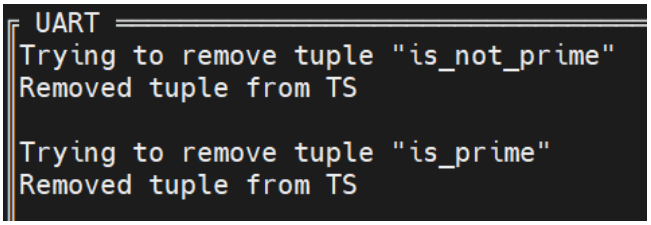
5. Opis funkcjonalności oraz implementacji Aplikacji Manager-Worker

5.1. Manager

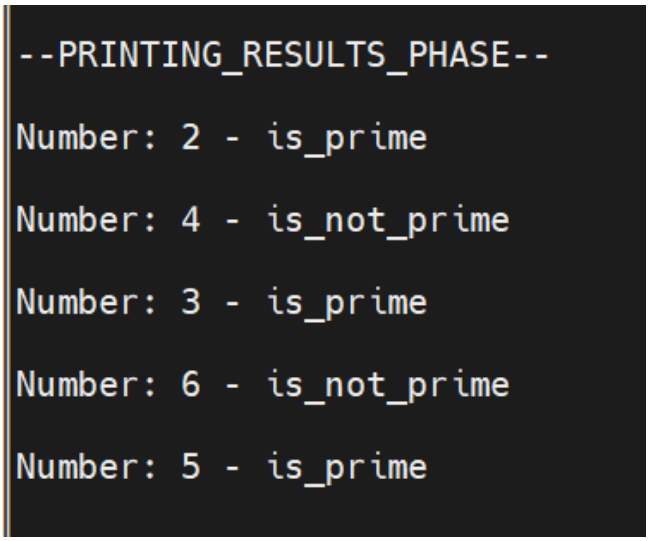
Manager zleca zadanie sprawdzania czy liczba jest liczbą pierwszą, aby to zrobić wysyłana jest krotka z liczbą do sprawdzenia.



Po przesłaniu odpowiedniej liczby korotek, zaczyna pobierać krotki ”is_prime” oraz ”is_not_prime”.

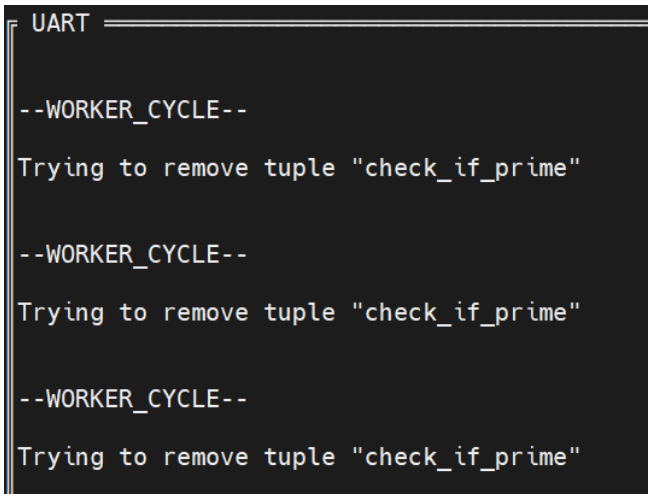


Po odebraniu wszysktich krotek wypisuje je na port szeregowy.

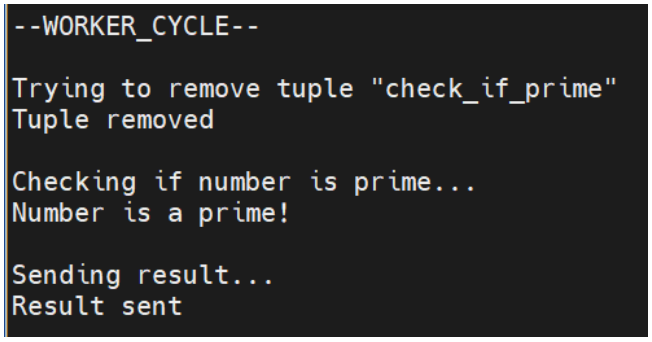


5.2. Worker

Proces Workera próbuje cyklicznie pobrać krotkę "check_if_prime" z przestrzeni krotek.



Po pozyskaniu krotki Worker sprawdza czy dana liczba jest liczbą pierwszą i zwraca do przestrzeni krotek krotkę z odpowiednim wynikem.



6. Podsumowanie

Udało się zaimplementować działający serwer, protokół ALP zgodny ze specyfikacją podaną w prezentacji projektu oraz aplikacje manager-worker sprawdzającą czy dana liczba jest liczbą pierwszą.