

Relatório

Bruno Tomé¹, Ronan Nunes¹, Matheus Calixto¹

¹Instituto Federal de Minas Gerais (IFMG)
São Luiz Gonzaga, s/nº - Formiga / MG - Brasil

ibrunotome@gmail.com, ronanncl@yahoo.com.br, calixtinn@gmail.com

Resumo. *Relatório final sobre a implementação das conjecturas de Goldbach e Collatz.*

1. Introdução

Este programa confere se as conjecturas de Goldbach e Collatz são verdadeiras até um limite superior informado pelo usuário, que no nosso programa pode ser de até 65 mil. A conjectura de Goldbach diz que todo número par maior ou igual a 4 pode ser formado pela soma de dois primos, e a conjectura de Collatz diz que sempre é possível chegar ao número 1 a partir de uma fórmula, se o número é par, o mesmo é dividido por 2, se é ímpar, multiplica-o por 3 e soma-se 1 a ele. Collatz afirma que seguindo essa sequência é inevitável que em algum momento chegaremos ao número 1.

2. Grupo

2.1. Integrantes do grupo

- Bruno Tomé - Matrícula 0011254
- Ronan Nunes - Matrícula 0011919
- Matheus Calixto - Matrícula 0011233

2.2. Líder do grupo:

Bruno Tomé

2.3. Divisão de trabalho realizada:

Como dito previamente, ambos os integrantes fizeram o trabalho por meio de Skype + um editor de textos compartilhado chamado Pages (disponível em www.icloud.com). Todos trabalharam e deram suas ideias para implementação do código, compartilhamos o código via pages a cada momento em que alguém fazia uma alteração, copiamos o código para o lazarus e discutíamos as alterações efetuadas via áudio no Skype.

3. Ambiente de desenvolvimento

- **Bruno Tomé:** Usou um MacBook Pro 2012 rodando o sistema operacional OS X 10.9.1 Mavericks, CPU Core i5 3ª geração 2.5GHz e 16gb de memória RAM 1600MHz. Compilador fpc juntamente com o terminal do OS X e IDE Lazarus. Ao fim de cada dia testava o código em uma máquina virtual Ubuntu 13.10 com as mesmas configurações.
- **Ronan Nunes:** Sim + intel core i7 3612qm 2.10ghz, 4 gb RAM, Windows 8.1. Compilador fpc, Lazarus
- **Mateus Calixto:** Dell com S.O Microsoft Windows 7 Ultimate SP1, CPU Intel Core i5 3337U 1,8 Ghz - 2.6 GHz, 4 GB RAM DDR3 - 1600Mhz. Compilador fpc, Lazarus.

3.1. Instruções para a compilação e Execução

Com o arquivo conjectura.pas no Desktop abra o terminal e digite a sequência de comando:

```
cd Desktop
```

```
fpc conjectura.pas -oconjectura.bin
```

```
./conjectura.bin
```

Quando o menu do programa aparecer basta digitar G ou g para a execução da conjectura de Goldbach, ou digitar C ou c para a execução da conjectura de Collatz, caso não seja digitada uma destas letras mencionadas o programa fará um alerta. Após a escolha, digite um limite superior de 4 até 65 mil para a conjectura de Goldbach ou um limite superior de 1 até 65 mil para a conjectura de Collatz.

4. Estrutura de dados

Na conjectura de Goldbach usamos o tipo de dados WORD e LONGWORD para as variáveis, pois WORD suporta os inteiros positivos até em média 65 mil, e LONGWORD até 4 bilhões, como só trabalharíamos com números positivos não faria sentido usar o tipo de dados INTEGER.

Já na conjectura de Collatz usamos o tipo de dados LONGWORD para que o limite superior de até 65 mil fosse mantido. Não pudemos usar WORD pois caso um número ímpar próximo a 65 mil fosse multiplicador por 3, estourava os limites do tipo de dados WORD, por exemplo: o número 64999 pode ser digitado em nosso programa como limite superior, mas, como é um número ímpar, a conjectura de Collatz diz que devemos multiplicá-lo por 3, logo, $64999 \times 3 = 194997$, estourando o limite de até 65 mil em média oferecido pelo tipo de dados WORD.

5. Algoritmos

programa Conjecturas

procedimento Goldbach

INÍCIO

Primeiro cria-se um vetor de números pares s partir do 4, através de um contador, e salva-o em um vetor até um certo limite superior.

Após cria-se um vetor de números primos até um limite superior previamente selecionado. Após a criação desses 2 vetores, inicia-se através de laços de repetição (for) a conjectura de Goldbach, que diz que todo número par a partir do 4 pode ser formado pela soma de 2 números primos.

O programa faz os cálculos até um limite superior e imprime na tela por exemplo:

ESCREVA ('Goldbach 4 ', vetorpar[i], ' = ', vetorprimo[j], ' + ', vetorprimo[k], ' (OK)');

Ficaria então: Goldbach 4 = 2 + 2 (OK) e assim por diante com todos os números pares seguintes até o limite. (6,8,10,12 ...)

FIM

procedimento Collatz

INÍCIO

A conjectura de collatz se define no seguinte propósito: Um limite superior e pré selecionado, e todos os números partindo do 1 são submetidos a operações onde, se ele for par, é dividido por 2. Se for ímpar, é multiplicado por 3 e somado 1. Essas operações são feitas sucessivamente até que o resultado final de 1.

Com laços de repetição e trocas de valores através de "SWAP" a conjectura de Collatz é implementada.

Ao final de todos os cálculos, imprime-se:

ESCREVA ('Collatz X -> (...) -> 1 -> (OK)')

Onde X é o número do qual serão realizadas as operações, contando do 1 até o limite estabelecido.

Exemplo :

Collatz 5 -> 16 -> 8 -> 4 -> 2 -> 1 (OK!)

FIM

var // Aqui são as variáveis do programa principal.

INÍCIO DO PROGRAMA PRINCIPAL // É executado antes de qualquer sub-programa.

Aqui primeiramente é implementado o menu de interação com o usuário, disponibilizando para ele a escolha de qual conjectura testar = 'g' para Goldbach, ou 'c' para Collatz.

Após a escolha da conjectura, há a escolha do limite superior. Sim, o limite superior é escolhido aqui. Pois esse pedaço de programa, sempre é executado antes dos procedimentos. Um procedimento só é executado quando, nessa parte do programa, ele é chamado.

Após a escolha do limite, um CASE ... OF diferencia a escolha do usuário, e então o programa chama o procedimento correspondente à escolha:

'g': goldbach(limite_superior);

'c': collatz(limite_superior);

FIM DO PROGRAMA CONJECTURA.

6. Conclusões

O objetivo do trabalho era provar as conjecturas de Goldbach e Collatz até o número limite definido por nós de 65 mil. Nossa maior dificuldade foi a dúvida de qual tipo de dados utilizar para que houvesse economia na memória utilizada pelo programa. Objetivamos um ganho na experiência em programar, já que o método para implementação da resolução das conjecturas exige maior atenção do programador. O tempo para o cálculo de números grandes na conjectura de Goldbach foi um fator limitante. Uma decisão importante para que a implementação ocorresse de forma rápida foi a não divisão de tarefas, utilizando edição de texto compartilhada via iCloud + Skype, onde os 3 integrantes do grupo participaram ativamente da implementação do código, possibilitando maior compreensão.