universidade de aveiro

deti universidade de aveiro

The Merkle-Hellman Cryptosystem

ALGORITMOS E ESTRUTURAS DE DADOS 2021

Prof. Tomás Silva

Daniel Ferreira – 102442 – 33.3%

Guilherme Antunes – 103600 – 33.3%

Pedro Rasinhas – 103541 – 33.3%

ÍNDICE

Introdução

The Merkle-Hellman Cryptosystem

O objetivo deste trabalho prático foi implementar e estudar o comportamento de diversas técnicas de resolução do sistema de criptografia de Merkle e Hellman.

Este sistema consiste num problema de soma de subconjuntos oculto numa sequência de números inteiros positivos em que a mensagem secreta é uma sequência binária em que os uns denotam a contribuição do número nesse índice para a soma desejada (a mensagem encriptada, disponibilizada) e os zeros a não contribuição.

Por exemplo, para um problema de tamanho *6*, em que a soma do primeiro e terceiro elementos da lista de números corresponde à soma desejada, a solução seria *10100*.

O trabalho desenvolvido baseia-se num conjunto destes problemas: 20 valores aos quais se tentam chegar a partir duma lista de tamanho *n*, variando de 10 até 64.

Abordagem

Recorrendo a diversos algoritmos, exploraram-se as técnicas computacionais mais eficientes e mais céleres para a resolução deste sistema.

Nomeadamente, utilizámos e comparámos 4 algoritmos de diversos níveis de complexidade computacional:

- *Brute Force*
- *Clever Brute Force*
- *Horowitz and Sahni*
- *Schroeppel and Shamir*

A *Brute Force* e a *Clever Brute Force* são funções recursivas que abordam o problema duma forma mais simplista do ponto de vista do programador, mas que aumentam a complexidade do problema do ponto de vista computacional.

A *Horowitz and Sahni* e a *Schroeppel and Shamir* baseiam-se na técnica de *meet-in-the-middle* que resulta numa drástica redução da complexidade computacional da resolução deste problema já que reduz o número de somas a efetuar, sendo essa a parte do problema mais laboriosa e demorada, tornando assim a sua resolução mais ágil e possibilitando a resolução de problemas com um valor de *n* significativamente mais elevado, face às funções recursivas descritas anteriormente.

Brute Force

Como referido anteriormente, a primeira abordagem foi escolhida com base na simplicidade da sua implementação, recorrendo-se assim a uma função recursiva. Em contrapartida, a sua complexidade computacional não é reduzida devido à falta de otimizações.

No método Brute Force, o *array* de números é percorrido e são efetuadas todas as possíveis somas. Para cada elemento do *array* e para cada possível soma, é gerado um *bit vector* em que um 1 representa a contribuição do elemento do *array* nesse índice para essa dada soma e um 0 a não contribuição, tal como na descrição do problema. Caso a soma obtida corresponda à soma desejada (mensagem encriptada do problema), o programa termina e devolve o *bit vector* correspondente. O programa também termina caso não encontre a soma desejada.

Durante o processo de execução, a função apenas descarta o subconjunto atual que está a ser utilizado para obter a soma (conhecido como **ramo**) quando esgotar todos os elementos ainda disponíveis e ainda não tiver encontrado a soma desejada.

Devido à natureza abrangente deste método, revela-se pouco eficiente, tendo uma complexidade computacional de $O(n^2)$.

Clever Brute Force

Como o nome indica, a função Clever Brute Force é uma versão otimizada do método Brute Force anteriormente referido, recorrendo à técnica de *branch-and-bound*.

Adaptando uma das funcionalidades da função anterior, o facto de descartar o **ramo** quando esgotasse o número de elementos possíveis de somar, este método expande o número de critérios em que este corte ocorre, diminuindo assim o número de somas a efetuar, levando a uma maior eficiência.

Nomeadamente, abandona-se o **ramo** atual quando a soma obtida exceder a soma desejada ou quando a soma dos elementos ainda disponíveis for menor que a soma desejada, ou seja, há mudança de **ramo** quando se prova que é impossível atingir a soma desejada com a atual.

Para além disto, o *array* é percorrido de forma decrescente de modo a aumentar a probabilidade da soma no **ramo** atual exceder a soma desejada, desencadeando o processo acima descrito e minimizando o número de somas a efetuar, contribuindo ainda mais para a eficiência deste método.

Como se trata do mesmo método base de tentativa e erro, ocorrendo apenas um descarte marginal de casos a considerar, a complexidade computacional desta função é também de $O(n^2)$.

Horowitz and Sahni

Esta função utiliza a técnica de *meet-in-the-middle* que procura reduzir drasticamente a complexidade computacional do *subset sum problem.*

O conjunto de números da lista dada é dividido em dois *arrays* **ordenados** (recorreu-se ao *merge sort*) de tamanho aproximadamente igual.

Feita a divisão, calcula-se e guarda-se os valores de todas as somas possíveis de cada elemento noutros dois *arrays*, também eles **ordenados**¸ um com as somas do primeiro e outro com as somas do segundo, cada um correspondendo aos iniciais.

Partindo do facto que a soma de determinados números dum conjunto A pode ser obtida pela soma de outros que por sua vez resultam nos números contidos no conjunto A, percorrem-se ambos os *arrays* tentando encontrar o índice dos elementos, que somados, resultem na soma desejada.

Encontradas as posições dos elementos participantes, gera-se uma sequência binária de tamanho *n*, resultado da concatenação dos elementos participantes em ambos os *arrays,* seguindo a lógica do enunciado do problema: 1 para elemento participante e 0 para não participante.

De notar que este algoritmo representa um ganho de eficiência tremendo já que reduz a complexidade, face à função recursiva usada inicialmente, de $O(2^n)$ para $O(n*2^{n/2})$.

Schroeppel and Shamir

Utilizando novamente a técnica do *meet-in-the-middle* construiu-se o algoritmo mais eficiente deste trabalho, o único que encontrou todas as soluções dos problemas dados, até n=64, sendo que o algoritmo anterior apenas atingiu o valor de n=57 com sucesso e os que se baseiam em força bruta, o valor de n=35.

A primeira diferença relativamente ao terceiro algoritmo é que nesta implementação a lista de números do problema é dividida em quatro partes ao invés de duas. Calculadas todas as somas possíveis de cada subconjunto, são ordenadas com recurso ao algoritmo *merge sort*.

Seguidamente, gera-se uma *min heap* e uma *max heap* cada um para dois subconjuntos, sendo os valores raíz a soma dos primeiros elementos dos dois subconjuntos e a soma dos últimos elementos dos outros dois subconjuntos, respetivamente.

Efetua-se a soma entre os valores descritos; caso seja superior à soma desejada, altera-se a raíz da *max heap* para a maior soma seguinte, descartando a atual; caso seja inferior, descarta-se a raíz da *min heap* passando a utilizar a soma imediatamente superior, ou seja, utilizando o elemento seguinte de apenas uma das *heaps*, mantendo a utilização do primeiro da outra.

O programa termina quando a soma encontrada for igual à soma desejada.

De notar que as somas encontradas são geradas *on-the-fly*, permitindo assim uma redução substancial na memória utilizada.

Devido à maneira ponderada como este algoritmo foi elaborado, permite atingir valores de *n* não antes possíveis, já que resulta numa redução da complexidade computacional para $O(2^{n/2})$.

Resultados e algumas conclusões

Para se ter uma melhor noção de como o tempo de execução de cada algoritmo vai depender das especificações da máquina onde está a ser executado, decidimos executar os algoritmos em três computadores diferentes (com as suas especificações abaixo descritas).

Nas páginas seguintes vão ser apresentados diversos gráficos a demonstrar e a comparar os tempos de execução para as diferentes abordagens ao problema. Optámos por colocar um gráfico, para cada PC, a comparar as **Brute Forces** e um gráfico a comparar os algoritmos com a técnica *meet-in-the-middle* (*Horowitz and Sahni* e *Schroeppel and Shamir*). Dispusemos também dois gráficos a comparar a *performance* de todos os PCs, um para os dois primeiros algoritmos (**Brute Forces**) e outro para comparar a *performance* no terceiro e quarto algoritmos. Por fim, expusemos um último gráfico onde entram todos os algoritmos executados por todos os PCs, com o intuito de se visualizar claramente as diferenças significativas entre os métodos brutos (**Brute Forces**) e os métodos mais complexos *(Horowitz and Sahni* e *Schroeppel and Shamir*).

Computadores e as suas especificações:

	Memória RAM	Processador
PC1	16GB @3000 MHz CL15	Intel(R) Core(TM) i7-8700K CPU @3.70GHz 5.0 GHz
PC2	16GB @3200 MHz CL16	AMD Ryzen 7 4800HS @2.0 GHz 4.2 GHz
PC3	16GB @2667 MHz CL17	Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz 2.20 GHz

Notas:

A falta de pontos nos gráficos do PC1 deve-se ao facto de o tempo de execução ser arredondado pelo wsl2 para 0 segundos, e, por conseguinte, o *MatLab* não considera esses valores para o *plot*.

No método Horowitz and Sahni o tempo que o programa leva a calcular os dois *arrays* de somas possíveis e a ordená-los é calculado apenas uma vez para cada *n e antes de estes arrays serem submetidos à comparação feita pelo método Horowitz até coincidir com a desired_sum.* Este tempo de somas e ordenação é somado no fim ao tempo médio e ao tempo máximo que cada soma levou. Decidimos proceder assim visto que seria uma perda de tempo executar esse processo 20 vezes para

cada *n, sendo que as arrays teriam sempre os mesmos valores,* o que provocaria um aumento dos tempos de execução.

Comparação Brute Forces

- PC1

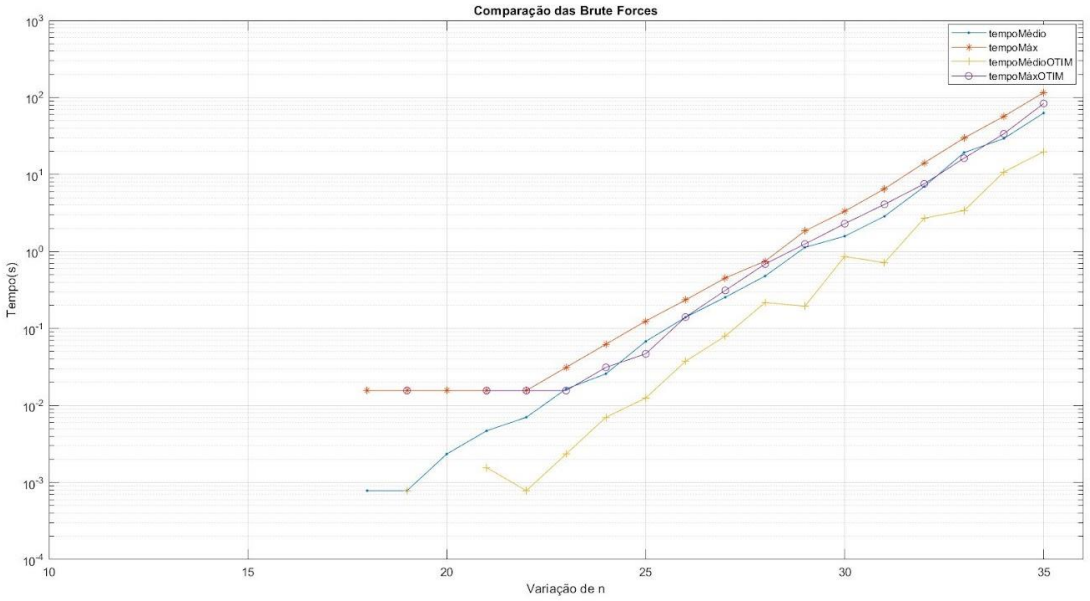

Figura 1 – Comparação dos métodos **Brute Forces** no PC1

- PC 2

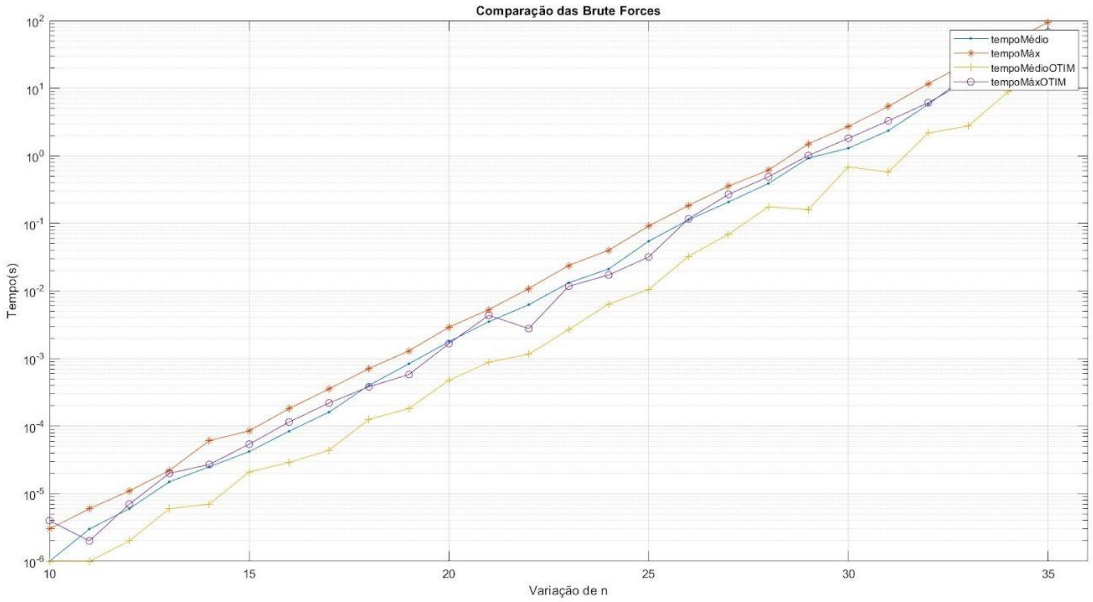

Figura 2 – Comparação dos métodos **Brute Forces** no PC2

- PC 3

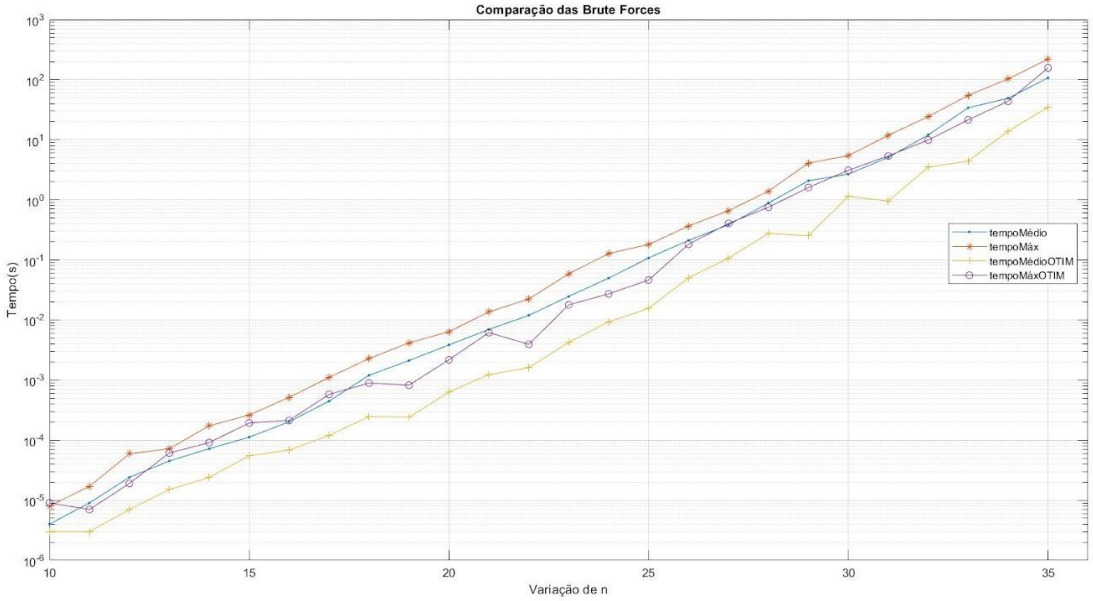

Figura 3 – Comparação dos métodos **Brute Forces** no PC3

Como se pode observar nos gráficos acima, a **Brute Force Otimizada** é significativamente mais rápida que a **Brute Force** não otimizada (como seria de esperar).

- Três PCs

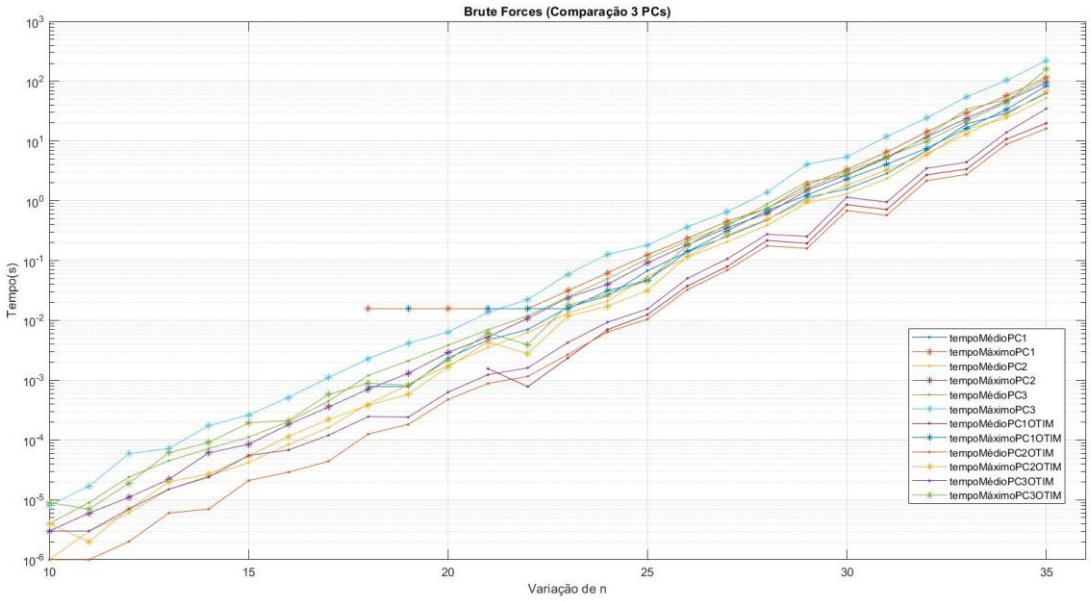

Figura 4 – Comparação dos métodos **Brute Forces** em todos os PCs

Comparação *Horowitz and Sahni* e *Schroeppel and Shamir*

- PC1

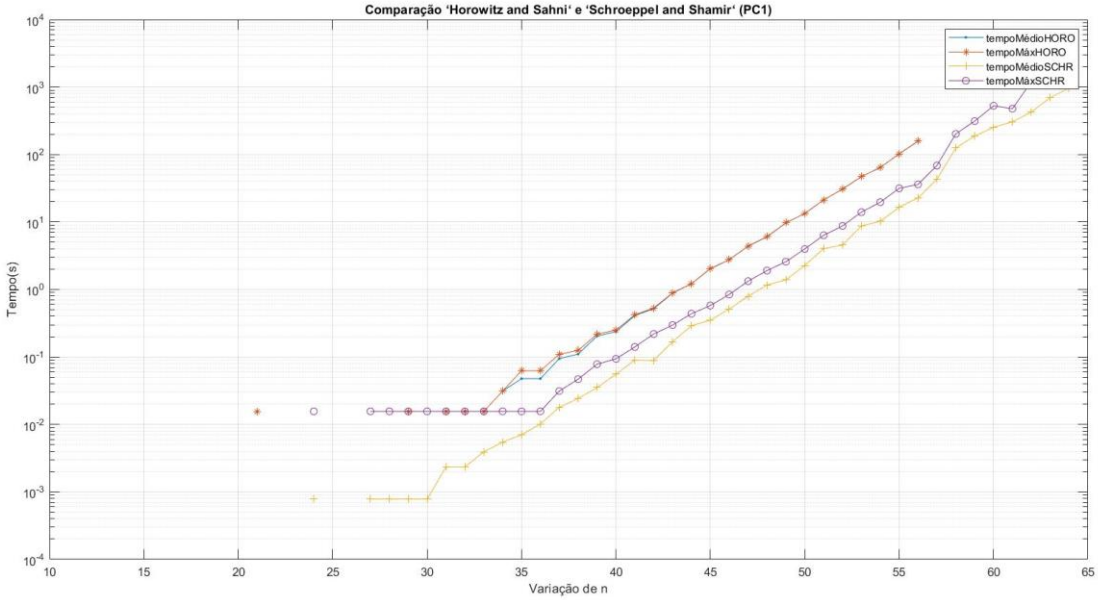

Figura 5 – Comparação dos métodos *Horowitz and Sahni e Schroeppel and Shamir* no PC1

- PC2

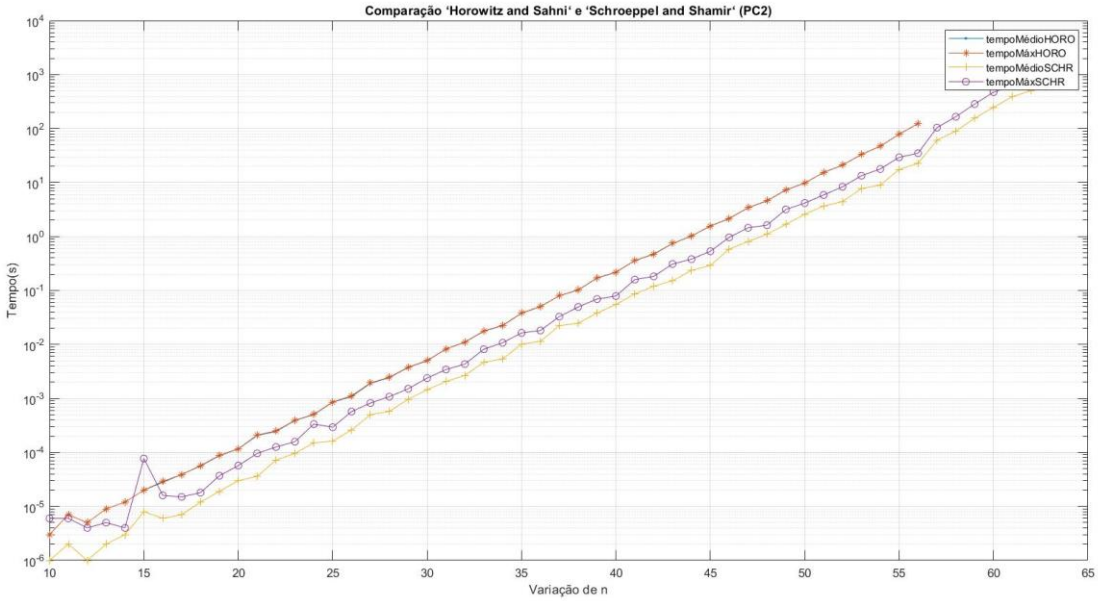

Figura 6 – Comparação dos métodos *Horowitz and Sahni e Schroeppel and Shamir* no PC2

- PC 3

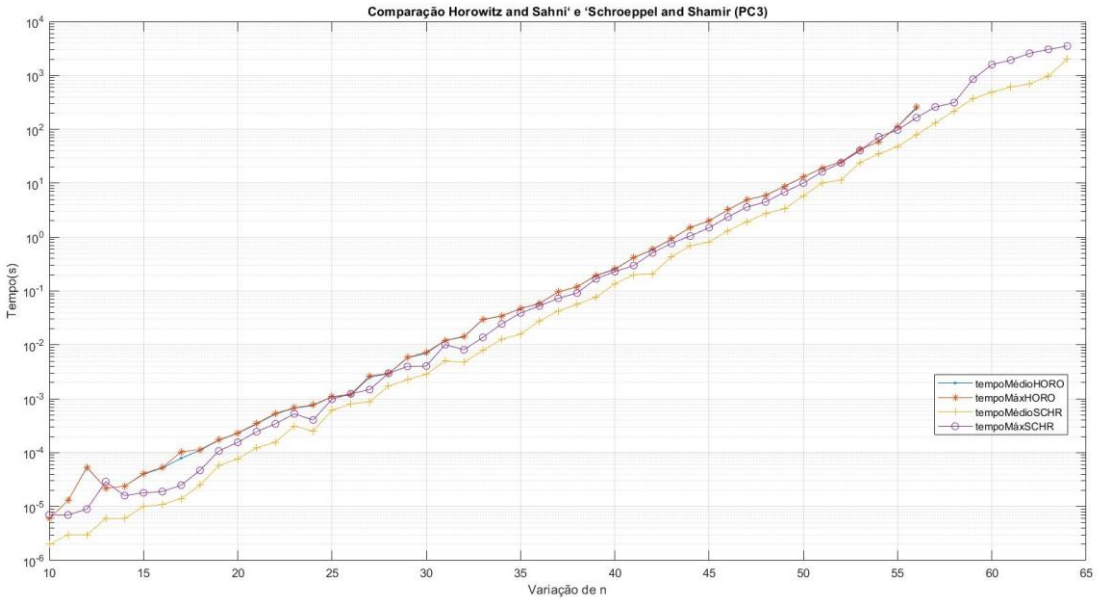

Figura 7 – Comparação dos métodos *Horowitz and Sahni e Schroeppel and Shamir* no PC3

Como se pode observar nos gráficos acima, a velocidade dos métodos *Horowitz and Sahni e Schroeppel and Shamir* é semelhante, porém este último permite alcançar *n* mais elevados.

- Três PCs

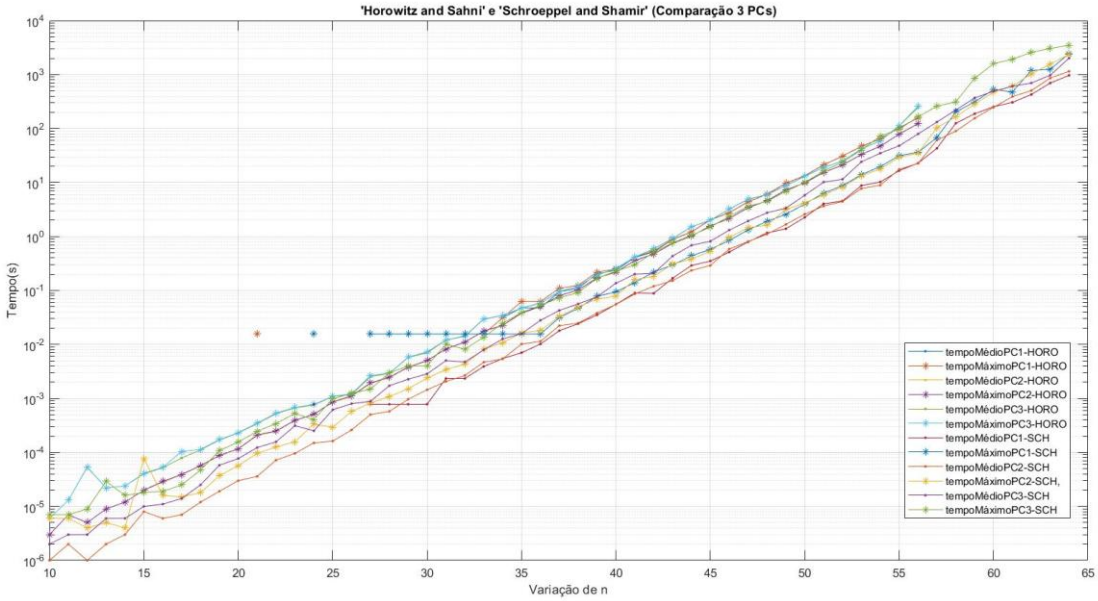

Figura 8 – Comparação dos métodos *Horowitz and Sahni e Schroeppel and Shamir* nos três PCs

Comparação de todos os métodos, em todos os PCs

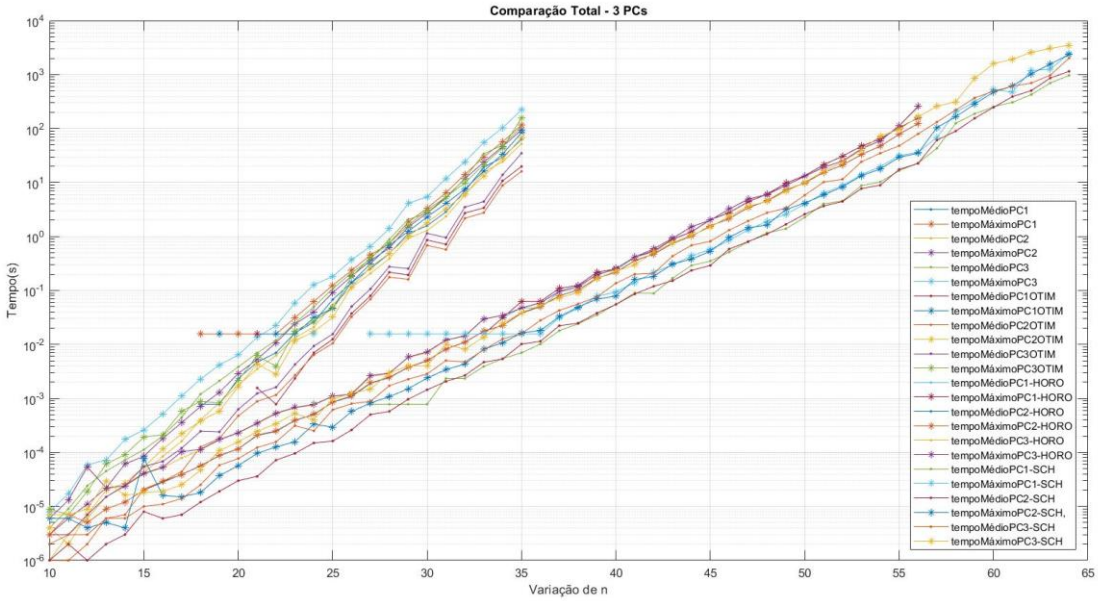

Figura 9 – Comparação de todos os métodos nos três PCs

Como esperado, o tempo de execução dos métodos *Horowitz and Sahni e Schroeppel and Shamir* é muito menor, quando comparado aos algoritmos de força bruta, permitindo também *ns* mais elevados.

Conclusões Finais

Em suma, este trabalho demonstrou que diferentes complexidades e diferentes capacidades computacionais irão originar tempos de execução diferentes e exigir tipos de abordagens diferentes, o que nos leva a concluir que na resolução de um problema é tão relevante a sua resolução como a decisão de qual abordagem seguir tendo em conta a complexidade do problema que nos é apresentado.

Soluções

Solução 102442_extra.h

0110000111
1011001111
1011110001
1010101101
0001101110
1100111000
1101001010
1000011111
0001110100
1101101000
1011010100
0010100000
0111111110
1101110011
1100011000
1101111010
0101111101
0000100111
0110011010
0111001111
n = 10

11111110100
01111010100
10101011001
11001000000
01001001001
10111011000
10111101111
10001000001
11011010000
00101001000
01011100110
11010110111
01111110111
00010011000
11111010101
10010001110
00010011111
00000010110
11111001110
10110110010
n = 11

100111101100
001100101100
001011110101
001110010110
110100100000
110100111001
001111111100
000011000011
110000001010
001101000111
010101111110
010111101101
101101000011
100011001000
111110000000
010111000100
111000101011
101101101010
010111111011
000111010001
n = 12

0101100100011
0101111100010
1101001010010
0001101001000
1111100010010
1111111100000
0011001111110
0110010110011
1101101001001
0011010000111
1100000101111
0101000001001
1101010011101
1011010110111
0010110010111
1100111100110
0111010111111
1000011111000
0000101011110
0111000011110
n = 13

01000001011111
10100011100011
11101111011110
01000000011101
00110111110111
00101000111100
11000010111000
00001001000001
01011100000111
11000111011110
10111111011110
11000101100101
01111011001101
10101000111010
01110111101110
00111011000111
00111001100001
10110101111011
10111110001110
11010111110000
n = 14
001011100010111
011111100010100
000111000110101
101100110001101
111111010100010
111011000011101
010000000000010
110101100101010
101100111011100
110100000001110
000111100100010
100110111100110
010000100111011

101100100100011
011011110111001
001110001001110
011100010110110
1110110111111101
1111010111110111
0011010111001100
n = 15

1010010001000000
1111101011011110
1110010011111101
1001100100010000
0101111100101001
1101011101110001
0110000110110111
0011001101110111
0101001110111111
0011110111001010
0011111010001100
1100011011110011
0010000111110001
0100000111001101
1011110101110101
1011001000000101
0111111100001110
1100100110010001
1001110110011010
1101101010010011
n = 16

1011001100111100
0111001111010010
0001000011011111
0111000101000011
0110110110010100
0000011111011101
1110000101110101
1111000101011010
1110011100110000
0001110011001010
1110000110110110
0110001001111010
0010101010011001
1010010111110101
0001001011110100
1000011000010110
1111000001000000
0111010100101100
0110001001011010
0111000111001101
1101011001110001
1001111110111101
0010101000101001
1001110000011110101
1110101101110101
n = 17

1101110110000011001
0101011000011000001
0010010101001111001
0111101011100011100
0100011010111010000
0111011110010010000
0000110110011111101
0001100011010111100
1001000010111100000
1001101111000011100
1100001001101111100
1000000001100010011
0110110011010000000
0110000110001011110
0110001111000011101
1101011001110001111
1001111110111111101
0010101000101010001
1001110000011101010
1110101101110101010
n = 18

1111100010001101110
0010010001110101101
0111101000101110110
1011100011111111101
1110011000001110001
1011111100000001100
0110001001110011100
1011011000001010101
0000110101111000100
0100100101100000101
1101001000000100001
0000001100101111011
1100100000000101010
0110111000101111111
0100111100101000111
1110000100011010000
0000001111100101011
0111000001100000100
1001110001111111111
1110101001011111110
n = 19

1110100111111000100 1
1000001010000101001 0
1011101000110101010 1
1001111010101110000 0
0110001100110100001 1
1010111111010100001 1
1011010111013001101 1
0011001100000001010 0
1111111110011000110 0
1010000000010111101
1010011110010100101
0100111100101011010 0
1010011110110100100 1
0101111011110101011 1
0010100010100000001 1
0100010001101011000 0
0001010110110010010 1
1010111111000010001 1
0110011110011101101 1
0000001101010001010 1
n = 20

0100110100111111010 1
1000011011000010100 0
0011101011110101010 0
0010011010100000010 1
0001101110100111010 1
1011000110010010101 1
1111001010010111100 0
1111101111011110001 1

011001101111111100011
001011010001111011110
001111101110101111111
100100001100111001011
001100011011110100101
111100110010111000111
011000111100001011101
000110100010110011100
100100001100110110011
111100010000000000111
110110111010010011110
010101111100111010001
n = 21

010111101101010110000
010101011010011100111
010011010001100111101
010000111110000000110
010110000100111110100
000001101100010101111
111111101001110001011
111100011111101010101
000101111100010110110
011010011100010011010
111111001111011010000
110101100111100011010
011100011110110110111
110100001111001110101
101011110000010010110
101111111110000111110
110000110011010110101
000010111001001011100
100111100011001000011
100111101101000000001
n = 22

111111110001110011111010
011110111010011111111100
110001101101111001010101
111000110110000011011101
000110101011100001000001
000011001010100110111000
001101001001011000010001
011110000110111010101001
010110000100100000010111
111001100010001011111110
000101101010010101110101
110001110101011000000010
110001101110110111101101
000110110011101101011011
000011000101001011001100
000101100011100000000001
001100010010001001011011
001000100110110110111011
111000111100100100011010
011111010110101100111111
n = 23

110101111000110110010011
111100011011000110100101
110100000000101011010010
000000001010101100110100
110101001000110110001000
011001000011010110111001
110010010011111000100110
000011101001000110101011
011011111010001010110011
010011010011100001000011
011111101100101110010000
001000000111111010110011
100111111101000111110101
100101001110100110110110
000101000100101100000100
011001011001110100100010
011110100001111100010111
101000111111101101100001
110010001010000100110110
100110011011001100111010
n = 24

001110001101001000111011
011100001100111010011010 1
100111001011001100110011 1
111011001001011110110001 1
000111000001001111111000 0
111001001110111110011001 0
010000111100100110000111 0
100111101111000010010101 1
110100101110010100011110 1
100100001001110111011001 1
001011110001010110110100 01
100111110100000000100000 0
001110101101000101010110 1
000101001000000100010001 0
000110100100001101100101 1
011010111000100100111111 1
110011000101000010100101 0
011010111000010010000110 1
011101001100010000111010 0
011010111000010010000110 00
n = 25

100101100111001101011010 10
000000001111100100111111 10
011011010101010010100001 11
110011101101111010101110 10
000111100000101000001011 11
100001011010010110000110 01
011001101011010110101010 0
001010000111001101110100 00
110011111111110011011011 0
100001101100101010100111 01
000100111100111010111000 11
101100000100111101010111 00
010011101011100010000101 01
010000001001010111001011 00
011011110001100011101100 10
010100100010111011110010 0
001100100010011011011100 11
111110110010010110001111 01
101111110010011100001110 1
111000111000101000111101 11
n = 26

111101011111100111011001 0001
010011000111111100111011 011
110001110001110100101011 010

011001111000000100001111011
101000010000101100011110001
100010100101000100110100001
000001010100111000101000011
011111101110011111010100001
110001110011101100111110000
110110011010010010110110101
100000000110101100011101001
010011000001100001010010111
101000011011110000101101011
011010100110000111001010010
101011110001010100100000100
011110110110111010011010101
111001001100100111010001001
110110011010100111000010001
100011010101101000111001010
000101011100000100111110011
111010111110010001101100001
101101010100011000001010111
101111111100101111011010000
001100111001001101101101011
101100101100010000100101100
101010000010101100101000100
100010100001000101010101001
111000110101001011110101110
000100011111011100000011100
011101101011111100100001111
n = 28

011111001100100011001110001100
110110111010100011010101001100
101100110010111010100101100111
110110111111111011101100111101
110100100011011100110111111101
010011101111111001001000100001
101001111111101010111111001101
111010110100101100010011111100
101110111111011010100010111
000100100100001000011101000001
010111111110001110011000111101
011101100000011011000011000 1
001010110010011110111011011101
001010011100101100001110100001
000000011000110001110110111100
110100101010100011100011011100
011010111100100000011000010 1
011110001011010111111010101000
100001110000111101111010100 1
000101010100010000100000000000
n = 29

010000111001001110010001110110
001011111010010101100001010000
100100110100100111001100110 10
000110010011010011000100110000
111111111001000101011100011 01
111100111010010100010111011100
001001100001100101010100000101
101010000000101101001000000 1
010001111110111001111110001110
010100000111000010110011010100
111000111110000110010111000010
010011010101110111011010010 0100
100010010010010111010110000111
110011011101111010000000000000
111011110101011100011111010010
000110001111101111001001110111
100101011110100010101100000000
001001101101000100001001110101
011010110101100110111010101100
011111100100011101000000001011
n = 30

100111100001111011000111011 1011
011001100010111010110111100111111
001001101011110011101000001 0110
100000000011101001001110100 10011
010001101000101100110010111 1111
110100010100100110011001000 11001
110100111100110111000111000 1001
100110001010101111100010001 0001
111001100010111001010000011 1010
001001110000011111100010111 1110
110001011110011010011010100 0001
000001110101011010110011101 1110
011000110000110110001010110 1101
100110000000101000110011100 0000
111000000011111111010110110 0001
011101001101011100100101100 0000
001101110000100111001110110 1101
100011011100000000100111010 1110
010011000100100000011001010 1111
010101010010110100110011010 0100
n = 31

111101010110010100110110100 110
000100001000001000100011011 10100
011111011011011000110000001 1100
100001100101101011100000011 10001
010011101110111000111100111 1011
100110010001010110010011010 00111
010001011001000011010011100 0100
111011000010000001111001011 10111
101101001010101000111100000 0101
001100001110010101110010101 01110
011111101100010001110001010 00010
011110101010111100100001011 01000
101110000100111000000001011 1101
100000001110010001010001101 01010
100111111101100000001110011 00100
111000111000011011000010110 11000
011111111110100000011011011 00110
001001000010100010011100011 01100
010110001001010000101100010 00101
001000111011001000010001011 01111
n = 32

n = 32

00100010111010110001111000010 1
01000001000001000111010100010 11
01001111001101010010110011001 100
01111110000001000010101011001 11
00000001100010100111010110110 00000
10111111100100100110100101101 011
11010010110010000111011010001 111
11110011010110001011100010111 110
01001100001011100101010111011 101
01100001001100011010001011110 010
01101010101110001101011010010 110
10001010110010010011010010100 0000
00010101100011101111000100001 0010
11001110000111001010100001110 010
01010100001110101101011001111 1010
01011101001010010011010010101 111
00000011110111100001000101100 0101
10000101011011001000101110110 0001
11110010110000011101001101011 1010
00111010010010000100111001101 1101
n = 33

11000001000011100100110010110 10011
11001000111011101111001100100 111
01101100110101100011111110001 00010
10000101001001100001000100010 11100
00001100111101110111001100101 01101
01101101110101010011001010101 01101
11110010110101111100011101011 011
00001001111000000000010010100 110
00000000100010111010011110011 00011
11100000100000101000011111101 0011
11001001011010110001111101100 00101
10111000010111000111110100011 1110
10100010001111000100010011000 10100
01000100110001110011000110100 100
11110111101100010110010111001 110000
01000000000010110011111010111 1111
00010010111001001000110000011 01101
01101100011100000000100011111 11000
00010111001010010011111110100 111100
11101111101001100110101110100 110
n = 34

11110011101000101111101000111 00101
11010100111111111010110000101 100011
10010101000111110010001001100 100000
01100010011001010110110111011 11011
00100110011000101110000110000 100111
11000001000000001100000101110 011101
11110001100010100111100000010 100000
11110100100000001100001100001 1010111
01101011101101000000010111011 1100101
10100010000011111001100010100 11000
00000100011100000010010101011 11101
00110010010010010111000100 1000000
01010101001000101010110001101 111101
01001101001000001110010110000 100101
01011100001111111111100101101 111100
00001011110101100110000101010 011010
10111010100101001101110011011 011110
00001000000001001011111101010 100111
10001101000101110110100110001 001011
00000110010110100000110101000 110110
n = 35

10001011010100001101111111010 110011
00111111111010000000010001011 0110101
11011000000011000011111011100 100100
11001001110001010010111011111 0110011
10111011100001010000100000010 0110110
10001110111110101100011001111 1001110
01111010011101001010110011110 010011
11001110111001010111110101011 111100
00100011111000101010010000110 1000010
00010100011111010111010010001 1101000
00001010101010000010100001000 1010010
01111000110100111000000011000 0001
11000100011111101001010100000 1111100
01011001011011010101111010001 000000
00011101101001001110111111000 0110110
01110011010111011011110000101 011111
01001110100000100111000011111 1001101
00100101100101110001101000100 1111011
01100110100110111000110101010 0100000
01010001011111101010100010100 100011
n = 36

01110000110100011100101111101 00100010
10101111101001000110010101101 1001110
10111000001111100001011010001 0100101
10000101000111110111110100101 1100101
01001100001110001100110011010101
00001111010101000111101010011 1111100
01101011001011001110111010100 0100010
01000000101000101111011101100 1110111
11110000111110111000110111111 1110010
11101010000111011110001001010 0100000
10100110110001001110110101010 0101111
01111111000111111000111000001 0101100
10111000100111110000110111111 0000011
11011111010100100000010010011 1001100
01110011010111011011110000101 011111
01001110100000100111000011111 1001101
00100101100101110001101000100 1111011
01100110100110111000110101010 0100000
01010001011111101010100010100 100011
n = 37

10110010001101110110001100011 10111101
10111000001100110100010000100 01101101
10100001101000100101010001000 01101000
01000011111100000010100011111 00101100
10000101000100011001011000110 01101001
11110110101011110001000011100 01101010
10011101000100110101010010111 01010100
01101101110111011011011011001 01010001
11110100011100011011011011001 10001001
01001111111100100110110011010 10101101
00110111011100111111110110011 00101100
00011101110011100000011010001 01011000
11101001011111011010110000101 01010010
00010110101000111111010100111 01011011
10100110100100010001100010001 01101100

1100110000011000111110011100001100101 0
0111111101111100011011011011001000 01
1010111100000000111011011000011010000 1
1011011010010110101010011010101001000 1
0010011010010011101001100100011010111 1
n = 38

1000000100011010001111110001101001101 00
0110100101110100110000001000010011111111
0101000011110101001111111001010111010 1
0000011110000000000000011110011110101 0
1100110000100111001010111110010001100 0
1111000110111110100100100011000101100 0
1100110101000010100001011100000011111 01
0101011001011110101101010100000011 0
1111001000101011000001110111001110100 10
0101011010001110010001101111110110100 01
1000011011000010000110010001000100100 0
1100010001110110001110101101010001001 0
0000011111111010101100101100110100100 0
0000101000000100011011011001001 01010000
0101110001100011011111011010001000011 11
0001010010100110101111001011000100001 1
0011010010100110111001000110111101001100011
1011001001010110010011101011010010010 10
0101010110000000010001000100001000001 10010
1101000000100001110011001010010010 10
1010011001011001100001100100100011010011
n = 39

1011100011000011100001011010001001110 11
1111001110100001011001100110100010000011
1100001110101111001100000010111000101111
0110010101101101100001011001001000011101
1100111000010011110001000110010001011 00
1010010000111011110010101100101100000100
1010101010000010011101011111111010001 1010
1101110111001101111011110011001100101110
1000010011100010110001001100011010011011
1011111010100101101110010000011111000101
0111011101100101101001110010010011110101
1001111111111011101100010111001100010110
1101010110101001100100001111000011010000
0111011111111101011111110001101000011100
0010001011110110001000000011100011000100
1011011111011101010001011100110000110000
0100000011111010000010001111111011110 10
0101001011000001011100101010000001101110
1010010010001011010111100000011000111000
1001010110001101010101001110110000100100
n = 40

0111001101011110001001100100001111110000 1
0101100111010110101000101101010100011100
0110100001110101001101100001010110010101
1010001011101101101010110100011111110 1
1011111101011110101100010000110001010111 1
0101001100010111000110010110100011111101
0010011011110110011100000000101011010 10
0011001101111001001000101100001100100010 0
1110010000000000100111100110101000111 10
0010000001000110010010001111001110110 10
1101101010011111011111100011011110000010
1110010011011100111100001010110011000010
1110000100111000101100011100111111011101
1001101111010101110111101011000101010001 1
0011010101110010111101010011101110010111
0011000000101111101010110101101100001 01
1010001101010100100111001000001100101010 1
1000111011101111101011110001000011101110001
1100011100010100010110011000010111001100 11
1001001001101100101110101101101110101010
n = 41

0110100100011101000111000001111001111001 11
1010100101000011100111101111011101010110
0111110011011100100110111011100100000111 0
1011011011011100001111110111111100101100
0000111010101001111110011001110100000010 11
0011101110110110011100000010000110011111 00
1100010010100011110110101011111110001010
1001111010111011100111010100101111000110 1
0010000001010001001011100010111011111000 101
1101000011011100001011000101011011101010
0010011100001111110011100110000101001001 1
0111011101101100110101101010000000010011 01
0101111010111110010011001011000001110111 0
1000110111011111100011011011001000001010 1
0010101110011010101011000011100100110111
0010101010001011100111110001011111111010
1011101111010011110111100010010000110111 1
1011101110010010101010011101011111111 11000
0011111010001010111010100100111010111100 100
0101110101001110001001011010011100111000 100
n = 42

1000010000110111101110010100100111010100000 0
1001101000101011001100011011111100100001
0010110011011100100000001110001011101011 0
0011100000010101111000100010001100011001 00
0010110011101110011111010000011111001111001
1100010110110100011100100101011110010000 00
0001110010101010000000011010100111101100
1000100010110101100100010111010010110011 010
1000000010111011001011100100010000010100 00
0001101100010000010100102011010000111001000
0010000011010011001000111010000010010010 01
1000011101110100001100000011010111111011 00
1000100011011110100100111111000010010111010
0110010010101000011100001001001001000100 011
0101111011111110100010010010001000100010 1001
1000011010110010001011011010100111001 1
1100101000001101111000110001011001111101111
1010100111010110101100110011000111011110 11
0110000011010011000000011011101111111011100
0011010001010011101111010010111010100 0
n = 43

0010011101110110100101110110111101100101 00
1111000101111001101111101001010000100000011
0101111011100001000110011001000111010111101
1010100111001110110011100001110000100011110 1
1111000101110110010011101011010101010101010
1100110000110101100011000000000101000000110
0001110000001000001110101000010100000111010 01
1010110011110100000101100101110101011100101
1111111000010001011010101100111001011010 01
0011010110011111000010011000100000010010 10
n = 44

00110101000110011111101111011101100110000110
11100101011111111100000100111000101001110
00001101001001000000100100110011011000001101
10110110001110100000101001101101011111000100
11010101001001010000001001001111001011011100
11011100011011000110101110011011010100010000
10111000001110100111110101001101100100010111
00001111101011010011110101100010001111000110
00010111110001001010111010101010111010000001
10101000101110010011111101010000101011111011
n = 44

00101100010011111001001110110000001010111000
11010110100111010011110001000010101000111010
11000110010001010010010011111001000000011110
11101110101000010101011100101000000011101110
00111001100010010101001001001010000101011000
00001110010011111001010011101110100011101100100
11100010100010011101001011010001111010100111
01001111101001100011111011001001011010000100
01111100101100111010000011101111000000001011
00111000001011011110010001001110111001010100
11000110110000010001001011110100101001100100
01111001100100001001001100000110011011001011
00000101001100011000111101101100001010110101000
00001111001010100111001101111010010001110000001
00001000000011010010010110000101011001110001000
01111101001110000000111111000000011111111001011
10110100000110010010010110000000110001011011111
00110000010101011100111101011010010101111010
10110001101010010101010000100000001110111100001
01001101110000011101100001101101111000010101110
n = 45

10100000111110100100110010001001010000001011110
01110000000010000011100001011011111011010101010
00111110010100100010010000011110000111010110000
01010101001010011101001010111001001110110001
01110000010010100010010011101111010000111010011
11000010101011001000011100001101010011100111111
01101101011100001111111010011010111010011010001
01011100010001111101001000010110110011100111011
10011010111011011001011111100110111010110011011
01001110001000111101001000000101000010011011110
11100001001100111000101001010101110011000110010
11111001000100101111111100110110010111000110101
10101000011001001001100000000111000000011011
10111111001111011111011011110111011011000010101
01100100001011110010101011101111111111000100
11101110011010011111011110101011110011111011001
10011110111100100101001011011001111011010110
n = 46

11111100000100000000011100100111000111110111011
11101001110101011010010100011111010001010100
01101111100000111001001000000101000010011011110
10011101011101101100101111110011011011110011011
01001111000100011111010010000111010000110101101
10100000010000000100111011100000000010000001010
00000001010000110100100111101011100110110111011
01011111100010011100001101010100110010010011010
01010001111111010011000010000001001001001111101
00000101000110110000111100111011010000011110110
11100000011010011111010110111000000110100111010
01110100110001010010100111101110001001011000010
01101111000011000001111000101111100000100111011
10100001011100001110100100111010111001110100010
11010101010001000110011001110010101000010100010
00000011010100111100100111010110011000001000
10100000011011100000101110011010010100010111
10111011011010000010001011101001110010110111011
11100011000110010010000100000110011011110100111
01001001110001100000011101100110000111110011100011
00110100111000101001011011101100011010010100000
n = 48

11001010010001001001000111001100010100111101111000
00111010101000011111010110101110101010010111000010
01100101000010010000000010010100101001110000001000
11111101100011111111010100111001100001101011110000
10010110001101110110011111010111000010001101111110
11110000110000011110001111010111000100011011101110
11010111111011011001100100100111000110001010111000
01011110011000111000000111000110001001001000111100
01111000000011001010111111010011000011010111111001
00111000000111001010111011110100110010101000010110
00000000100100100011010000100010010010111100
01010001000001110101100100010100111000110011000
01011010111011001010010100010111001100111000111010
11111011111101101101001010111000010010000101010100
00110100111000101001011011101100011010010100000
n = 48

11001010010001001001000111001100010100111101111000
00111010101000011111010110101110101010010111000010
01100101000010010000000010010100101001110000001000
11111101100011111111010100111001100001101011110000
10010110001101110110011111010111000010001101111110
11110000110000011110001111010111000100011011101110
11010111111011011001100100100111000110001010111000
01011110011000111000000111000110001001001000111100
00111010100000111010110010000101001110011010111010
00001100011100101010110111000100110011100010100
01001001100001001111000101000100101011101101100
01100011010100111000000100001101111101101000110
01101110011110100100100111000111000010011101111001
00001010001100100100111001100010010110111010111
00010110010000111001010001000101011101100010111
01000001100011111000110010111011111001101101110000
00001011000000000101101100000010100100111100100
01100011100001100110011010010011111010000100010
10101000100000011111010110100000011010111101100001
01100010111000010110000101001100101100110000101010
n = 49

10001010011000000110111010011100000110110110000101
11110111111010110011010100000101111010000011110101
10100011001101111101101101010000010110101000000
00011010010011101100100111000110010101101010011010
00010110111101001001111010001100101101011001010010

00010011110100011111111010001101101011110001111011
11100000011100010000011101111110110100101010001101
11110100111010001101001100101000000010111110111100
10110100011101001111100111010101001110100001110101
10100010110000100110110011010110111101010000101111
11010110000110001100011010110100010111011011111001
01110011010110101110011111111001101000010000100101
00010010110001110011011011010010001000001100000001
10001011110101000111000111110100111101101011011110
00110011100111111010110011010001011000110100001110
10010101010110001100001111000100010000001000010000
00110001001101000010110010100101011101111111101000
11111010101001010011100101100010101001001010001101
11100010100111001110101111111100000000100100000010
10010000100100111101101111011001000100100011111011
n = 50

0000011011111100101111101001000101110001110110011100
110011111000000000110010010101100010001101000110001
000001001111010111110011110100100001111101011100000
000111000100100100100111001000110000011010110111011
101111011000001000110111001011001010001110011000010
0101111101101100101010000100100110001101100101000100
100000101101011100010000111111001000010111000111010
100011010011101101010011011000000101001001000011100
011011010110111001001010100111011100111010111010010
101010000100000000000011101000010010000000011111010
111011110001101100001011000101110000100011110110110
100000100000111000001000100000000010011011011111110
011101000101100101101010000101001001111010001101011
110011111100111010100001100111010101100100000000110
100000011111110011111111101110101101011101011011111
000111110100001111101000011010100001100001101001100
011111100100100010011000001010100111011100010100001
110010011101010101011011001000010100010000111100111
100110100110000111110011100100001000000010010001110
0010000001000001101110101111101111000010000111010110
n = 51

1110100001010110111000000111011111101000011101110100
0001111011110000100111011001010010000111110100101011
1110010011101111111101010010010110110100011100111101
1010101011101101100000101000001000001111111111000011
000011100100011111101000111101000000101010001100001
0010000000111000000001000010001010010001100110000000
10001111111000101001100001001000100111000101111110111
0111010101000001101011010001011000000010111111111000
1111110110100110101101111100000011000011000000011010
10010001011101110001001011001100001010100000010110101
0101011010101100111001001011001000011001100000111000
1010011010010001010010111100101100000010100010001100
0000111100100110100100111001100101001000001000111110
1011011111110001111011101000100010000100011010011101
01010000101101100000000010000001111010101101101111010
0000101111100111010110111001010100101010010101011111
1010111101010010100011011001001111100001011110010000
10010001011000001100101101000000101101110111110100111
01100101000000010110011111101110011000011111101100100
1101011110111010111100001010010111101110110100111011
n = 52

001100100111101111100010111111101011010101011010001100
0010111011100101110111100011110001101101010100111001
00010111100111000000110110000011001111000000011000111
101000101100001010010101010110011001000000001001100000
11010101110000101101101110110011101101011111001000010
01001011101101000001111101011111111010101101101100111
101110000001101100011100110101001101100000100001100100
10000000110000011010101000101011000001111011011101001
01010111111001101110000100011011110000000001101001101
011001101010000000100011101011001000100001001111010100
011000111101111001001010101010011011000000000111111101
11100001110011001100110110000100110110010001011110111
10111101100110100000101001110100110111101101011010000
10000001111011010000010010100100111111011000100110001
10011000001011010010010110111101100110100100101100000
01110010010100000110010110001111111000011101001111000
11010000110101100001001101010011010110111101110010111
11010110111011000101011110001010100010011001110010010
00010010110011100010000011010110010011000010010111010
00010111111010001110010110001101101011010100110000100
n = 53

101101110010011000000111101010001100010101110101110110
000011110100100001111100010011101101110000111111101111
010111011110000111001001111000101110101110000100111010
111101001010010110100011110110010100101111110111100101
000100100011010110001111100000011111100111010000100110
000100110111001100100111001110010111101000001110101110
010001000010010011101110110010000110001011111001101110
011000011001100100000000101111011010001110000001101110
1010111100111100010001001001011011111100010110100011011
011001010010101001110011101101111000110011100001100001
0101001011100000001001011011001111100111110111110010000
001101011001000000100011001001011110001001101100011000
011001111001101000110001001110101011000111110001101001
100101000010001000111010011100011100100001011000111100
001110111101011000100011101110010101010111100110111111
010100000110110101010110101110001101111010010110101100
101001110110010010101110010101011000010011110101100011
1010101000110011010101111111110000011111010101100101111
001010000010101001111000011010010110100101101000111011
000001110111101110000000011110101010111110111000011011
n = 54

11011111110001100010101011101110010111001011111001100000
10001111000000000000011001010011111011000001000000100011
0110111011110011110001011100110011110010111000001100011
0111001000001001110011001010101001111101110000100101010
0110101010100111001101111010010000010011100101110100101
1101010011100000110001001101100100001110001011100010110
1100000110010010010110011100011011110010100010110100110
0001110110010011100001110111110011101011001110100110000
11011011011001111000111001000000100110110110010001000000
11010001010101000000010101000011001010010110100011000111
1111101110001101100010011110110011111101000001111000000
1010000101111101101110110110101110101110111111010101110
10001000111101101001000010100010000000010101110111010101
0001001111001111001101111010000100100101011000000110101
00000100001001110111111011110100010100110100010100010010
0000000100100011010001011010111010000110110001101111001
10110111111011000001111000101011000010000111000110001111
11110101110011011010000000011000010011010111111101001010
0101100110110111001010101010100101011010101000010101110
0011001000111111111100111101101011100100000001011001000
n = 55

11111100101111001011101001110011011010000111001101100110
00001100100111100000101110111110100111101111001100101001
01011011011001000110110011110010000010011110101000100111
1011100001010010011000110111101000010001100000000001010001
11111101000101101100110000111000100111000001011011111110
10100110110000000101000101111010101001001000011101000011 1
00101111111110101000001011000000101000000010001101011000000
01011001110011011000111100010101100111110010100000111101
01111000001011000011101110101001001110100010001110000000
110110110100000001110010011101100101111000100011010100010
11111010101111011111010100100100111100111100101000011110
11001110100001011110000111010100001100011001001100100010
10001000111100011100001100111111010100011010010111111110
11110011001111101101110111011101001111111001111110011100
11111000010000010101110111010100100100111001101111100011
011101100101110000000011011000001110100110110100000001011
00000101000010100010001111001000110110001011001010111100
00101001101100010110110011100000011110110111010011011001
01110001110100010110111101000010010101011010101100110100
111100100110110011101000000010101110000001100011100100110
n = 56

10011010001111110000100111111111110101010100110101111000001
1010100110010000111111000100000011011011100011110010100001
11010010000101011001110000101010011100011010001010101110 0
01111010000011011100011110101111000100011011010101010110100
110101110000010101011011000111101010000100001111100111000
110101111101010101111011100110101101101110011011110011010
111000001011011100110110001111100110010101100110011001100
011010101010011011001101110110001001110011001010011011101
110111010101011111100011100010101001100111001010001110011
0010011111101010011100111111011011000100110011101001100110
00000100101010111000010110111110011100111000011111000000001
110001010001110100001011001100100011001011001000011101010
001110000001110011110100101001001011000011011001000111001
11001101101101101010101011100110011000000000111001100101101
01001000000000101011000101000000110100000001101100101101001
100111111110010010100111011100111101011010010010100010001
100101111010100111100011100101000001010011110001110101101
111010111111110000111101010101010111011001101101011101111
000100011000100000001110010000100111101101111001100001100
110111111010110000010111111011000010010001110111010000100
n = 57

1111001111000010110000011100110000010010101010010100001010
0110111000000010010111111111111110110101101011100000111101
0100110101011101110101110110101111011010101100011110100011
1111101101100101100001011010110110000010111110001100101010
0000101110101011000110010101110010011001101011111101100000
1001011101111101000001011001001011101110110010000001000100
0111001101010100111100100010100000100111111101101101110101
0010100000000010001010111100000000111001111010000011001000 0
1001011111111001100010011011111101000110101110110001010101 10
0100100111000101110110001100011000001100011011111010111001
0000010000110100110100000000000010011011011000011010011110 0
1100111011011000001010111101000101000111001001001000010000
1110010110011001001010010001110011100010111100111001111011
1010011010100101110001010000000111001110010011000100011101
1011011001100101101111010010101101100110111011001100100101
0011011010111100100010010000101111000110010010001100001010
0101100010000110100010101010101100101000000111011011100010
1100011110101101011111101110001011001111001111110001011 01
0101010010000110001110100011100010101100000110011101010010
0100110100001000000000000111100010110001000111111000100000
n = 58

0011100100101000010010000100101001000011001110110100001101 1
1101101110101110011110110100100101100110100100010100100110 1
0001001111100111100110000011000100110011011011001101100001 1
0111001010100000000101100000001111111100110001010100110100 11
0010111011011011101101110110101101110011110101100100101101001
1000110110110001010011010111110010000110000011111111110011 1
1111101000111110111111010100010011110111010100100101001100 1
1001110011000111001000010111111100010001011011000000011110 0
1011000101000100010110110110100101001111110110110110110110 1
1011100000110001011001010001111000011000101101001110110000 1
0110110111011000000000010011100111010101101011111011011001 0
0011101111100100110000010001100101111100100101011110010101 1
0000000110111101110011101100101111010100100011000001011010 0
1001010110000111011100000101111110010110010000110010011101 1
0010101001001010010011010101100100101101000000110001110010 0
1111001110000100101110000110110010100010100000001001001001 01
1111010010111011101001011100011110011100111001100011110011 1
0001000101000101111010011011101010001011011001110010000000 1
1011101100110010101110101001111100100010100001101011101110 1
1110101010100010111001000000100001001110001111100111010110 1
n = 59

1010110111111011111011000011100001001111000111110000100001010 1
1100110111101001111011010010000000111010000000111001110110010
1100011000101010111101110111001001101010101111001100011001 11
1000000100110000111110111011010101111101001011111010110110 11
0000100011101100100001110110100111110110001101100100111010 10
0011111111111110100101000100110110011100101110011010010011 011
0101110111111011100111011110001101110000011010100111110010 111
0111010001110111100000000111101100001010110001010100000101 101
1110000110101000000101101000010111101011010111100110001101 011
1011010101100101101010101001111110010011101110101011111010 00
1000110110000100011101100100000000101110010100011110011001 00
0010101100001111011000010000001100000110100000000111110011000
0111001101011110100100010111101100001010100011110001000011 10
0101011010111100111010001001110100000011001100111101111011 11
1001110101000001111011101111111011000010001000000010001110 0100
1101001111100101101110110110101101011101011111100111100111 00
0101000010001010000100000111111000110110001000110101010110 11
1111101011100101100100101010110000000011100001000010010010 10
1011001110111010100111001000000111000000100110011010110101 111
0011100000000100110101101100010100011001000011011100110110 000
n = 60

0001001111000010010010110001110011000011100111100100011000111
0110000111011111111001110011000000101000100100100110110000000
1101110000101110101011101001001001011011001101010001111011101
1011000110011110111010001011011010100110100110010101010001100
1011110001000110101111001110011000100000100011111111110000010
0101100001011010000111001100100111101011011011001001010001111
1101011100101011101000110111101000101010100111111100000100000
0111100101011111111111001011100111101011100110010001111000000 1
0110010011001011011010000101100111000100011010100110101101101
1001000011101100101010100001010000111100111101001110111000111
0101000110101101011001110110100110000100100010101110110110100
1001100110100000100111111100001110110010111001010000100101 01
0111001011010111000010011110011111111011101000000100111101000 11
1001111100000111001000011110011011001101011111001001000110011
1001010101110111111000000001110000100101010100111110110110110 10
1110000101001110110100110110100000111110101101001011101100010
0010110110001000111111101010111100110001000010110110011111111

0010111011001111001101000011001101000000011000000001010111001
1000001000111110000011111010010100011010011011010000010110010
1110010000011110101110101000000101101110100010101001010001000
n = 61

1111100000101111001011001111000011100000100000111101011111110
1110101100000100001100000010101111010101010100111011001100000
0011010111010111001111011101000011001100101001000101011001011
1110111110001110101000101011111100000011010011110001001000101
0000011100100111000110011111101110001100101000011111011110101
1100000000001011101111000100011111010001010000110011101110110
0111010101110010110111010000000100111100001010011101101110010
1001100010110100100001010000011100010110010001111010010011010
0110000011110101000101100101010111001010000100010011101010110
0010101110000101011101000010110111010000101111010010011101010
0010111010010110101001000111100101100101001010110110001110110
1100000011001010101100011001000111111000000010000010011101101
1110010110110111000011110011001110000010100110010101000000011
1011000101110001101000001111010101001000111101001000111010101
1101101000111001101000010100010101011001010000100010111010110
1001011011111000101100011100111101000001000100001011110010111
1000011101100001011100001100010110111110111111101001001110010
0010111010010101101001000111100101100101001010110111000111110
1100000011011001000011111111001111100100100111100001110100111
0111110001110110011000100010011010101001000111010010010100000
n = 62

1100001100110111001001001011101010111111100011010100000100
1010100000100011011011010011101111000001001001111010011100010101
1100100000001001001100101000010100001001100110011101001100101000
1111000001111010100101110111111101010000001110111001110001011
1001110100110111011010000001001001001010011111001010100111000
0010011010011011000001011011010000101111001011010110111001111110
1100001110101111101101111110011101101010110000100101011100011000
0101101000110001001001001000111101100011111010101000010010111000
0111001001001000100000011011111001100111001010000111010101111110
1001100110110010011110100100010110111111001111111110000100111 0
0111110101111000001110001101111100001110011010000001111110010110
1101101111000100000110001010011110111001111000011110001101000001000
0010101110100110011110000010101001111011101101000011011011110
0111100100111010000110010011100100001110101010010011010001111111
1101111101111110000111000101101100100001110100011011011100010
0101000100111001011011010110111011001100101010000110000011110 10
0101011000011111110011011001010100111011001010100001100000111010
0010101111110011111100100010001100001100000000100010000100010
1111010001100011011011100001000101011010100001010011010001100111
0100001011011000011010111011001000101010100001011011011000010

n = 63

1101011010110001000011100100000111001100101110100100110100011 0
0101111000100000011110001001001001011011010100000101100101000100
0011110100001100110011101000110110000101001000011101110010110110
1110101111110001100011000000001100111001100100110110110000000100
0000011001010100011111101100110000110100000111001010110000000100
1111101101011101010010001011101111000001011000100101111010 110
1110101111010001010111011000100011001011011011100000011111111000
1011111111010111000010000001000110010011011111010111111010010001
1110101000001110110110101010000011100101101110011111110111000010
0001100011101011101100100001101111000011001010011101110011100001
1010110111101110001101010001111011001010100010110100011111010011
1011000111000100001100000011011100001111010010001001110011000110001
1000010010010010001010010001101010001111001111110111000011100001
0010101101101010001110001101010100010001110011010011100111000101100
0000101110010101011000101011010111100010001100101010100101011111101
0001001101101000101010011100000011100000100100111011100000101001
1001111100011111011010011100101110101000010100000101111001111101
1101101010000001111001101011110101010001101110010010101010110
1111010010011010101000001110011000111011110111011111011111010111010
0110001101101001101010001100000011110111001110001111000000011000
n = 64

Solução 103541_extra.h

0110111110
1101111100
0000011000
1000101011
1111001000
0100001101
0001011110
1010110111
0000110110
1110111100
1110111101
1011011001
1100110011
1101000101
1010100101
1110011110
1011101100
0100011001
0000011011
1011101010
n = 10

11001010001
01000001101
01110011010
10001011110
10010110111
01011000001
10011110011
10000101000
11101001101
01111010011
10100101111
00111111000
01100011101
10011011001
01010010011
01110101100
11010101010
01100001111
10100001000
01110010100
n = 11

111001000001
100000101011
101111100110
100010111110
110110100001
011000101011
110101001101
000001110011
101101110010
011101011100
111111100011
000001110010
000010111100
011101001000
101100011000
101101001100
100101110011
011001001010
100000110110
011101000100
n = 12

1111111001100
1110100110001
0011110111011
1010111000101
1010110100001
1000010011100
1010110000001
0110001011000
1001101110100
1101001110001
0110110001110
0010000010110
0011010010010
0110110111001
1100010000111
0111101010011
1100111110001
0100110010100
1010101001010
0110110011001
n = 13

11000101100011
00001101010100
10011101000101
10001011000010
10101101011110
10110010101010
11011101111001
11010110011010
10110101111101
01110101010001
10111100100100
11000111001010
11010100100100
01010101100010
00110100001001
11111001111101
10101111101110
10111010011000
00101101100100
01111011010111
n = 14

1110111111111111
1000111111000011
0100001000010000
0000110110100000
0110110111110010
0011001101100011
1011100011011100
0110010000010100

000001001111000
111100110011011
101000001110000
000101011100110
101001100111100
000011100000110
011111100100111
101010000110110
110000001010000
100100101011000
010100010110000
101010100010100
n = 15

1000110010101110
1101000111011010
1101011111111010
0110000111100100
1010010111101101
1001010001000011
1010110001001101
0100011011110010
1111011111001011
1011001101101100
0000111101101010
0101101010100101
0011000111000110
1000011001010001
1111000110001001
1000001110000000
0011111110010111
0100111111000110
0100111001111001
0111010110000100
n = 16

0101001001101011 1
1110011100011010 1
1001101110011010 1
0101001111110001 0
1110011100100001 0
1101000010101100 0
0000011000000000 0
1010000100100100 1
1001110000100111 0
0111010000000111 0
1001001101111110 1
0001011010100111 0
1011001111101101 1
0001000111001000 1
0100000100111100 1
0100111001111001 1
0100100101011010 0
1100110001011100 0
0011001101100111 1
n = 17

0110010101100111 1
0001011010101011 0
1110100101110111 0
1011001101100011 1
1010100110000001 0
1110011101000100 1
1000000111101001 0
0101001110100000 1
0001111000110010 0
0110010011000101 0
1101100111010100 1
1111100100010111 0
0011011000000110 1
1000011110111001 0
0001011011001100 0
1100111011001111 1
0010010011011011 1
0111011000100001 1
0101110011000110 1
1110110110011011 1
1010101000010001 1
0011001101100111 1
n = 18

1010101000010001 0
0011100101100000 1
1100111110110011 1
0110000001101001 0
1011010000010011 0
0110010010001000 1
0101000101101011 0
0101110011000110 1
1111100001001010 0
1011101111111011 0
0000011001011000 1
1111111111010011 1
1111111110101011 1
1110000010100010 1
0101010110011100 0
0111111110100000 1
1111000100001011 0
0010001001110110 1
0011000000100001 0
0000010000010101 1
n = 19

1110001011011100 0
0100001000101110 1
1001000101110000 1
0100111111010001 1
0111100110100011 0
1101001100110010 1
1010000101111110 1
1011110100111100 0
0001100111000111 1
1001100110010100 0
0101110110110100 0
1000001111001010 1
1110011001111111 0
0011101100010000 1
1000010001011011 0
1001101001100100 0
1111111110110010 0
0101010111000111 0
1011010111110010 1
1001010011011000 1
n = 20

1110101011011111 1101
1000000110001111 0010
1011011111110011 1110

100001101000011111100
010111111100001101000
000100110001000011100
111011011001100011
010110111011011111110
000000110111001011100
100010110011010101011
110101010111110100100
000101010000011010000
110000101111011111111
011011110001111010001
011101100000011011110
000000010100010100100
111110000001100100101
100111000011010101111
101101100111101110101
100110111110111110100
n = 21

010101001111111010111100
011100101000111100101010
110010010000110100101010
111001110101110000101011
010001001001000000010111
101010110100101011000001
000011100011100011000001
110101101101011111010011
011111001000101101011000
111010110010010101010010
100010000010001100100001
000100111001001110000011
011101111010111010000000
101010001110101010101001
011001001011011111011
100110111001011010111110
101010110100101110101
001011011101000010010110
101001001101011010010
011100001110101110100
n = 22

101001110110111010100000
010101111101011011111111
101001111110100001101111
110111011100100010111011
101000000101000001100010
110011110111100000101110
001011101011100010111011
110000001011101011100000
001001101101101110000110
011000110111000001111111
110011001111001100000011
101101110011011001000001
001110011100111100011000
001110100001010000011111
010000011100010000010011
011011000010100100011111
000100100110110101110111
111011000111101101010100
010111010101100100100010
001110000000111001101011
101110100100100111010010
000011101001000111111100
111101001110100101000100
010101111010010111101100
010101110000101000110101
111101100110110110000000
101001101111110000101010
100010001100110010100010
n = 24

110110011000111100111000 1
100111100011001000011010 1
011100111000100111111001 1
011000011000000010011111
011010000001000110100101 0
001100111000010111011101
011011001100101111000011
111100110011111100111011 1
100101000110001100101011 1
000100001110000101100110 0
111001000011010111011010 1
010011110010010000101100
111111101101011110111010
000110010110001100111010
001100011000111111111011
100110111010101100001010 0
001001011110000100011111 0
100010000001010001000111 0
100011010101001011010100 1
010110001000111010110011
n = 25

100000000011100110101011 00
100111010100110010110100 00
111001101100001010001111 01
111100010100110101111011 10
010001101101001101010111 00
111011000010110001001011 01
101110001100010100101011 11
110100000001011100100101 10
001101001010111001110010
111010110111111110000100 0
011000001100110110001010 00
101011100010001101010000 1
000101000111010001100111 0
000000110111010001111111 011
111010000000101111100110 10
001010011100001010010111 0
100010000001010001000111 0
100011010101001011010100 1
010110001000111010110011
n = 25

100000000011100110101011 00
100111010100110010110100 00
111001101100001010001111 01
111100010100110101111011 10
010001101101001101010111 00
111011000010110001001011 01
101110001100010100101011 11
110100000001011100100101 10
001101001010111001110010
111010110111111110000100 0
011000001100110110001010 00
101011100010001101010000 1
000101000111010001100111 0
000000110111010001111111 011
111010000000101111100110 10
001010011100001010010111 0
100010000001010001000111 0
100011010101001011010100 1
010110001000111010110011
n = 25

100000000011100110101011 00
100111010100110010110100 00
111001101100001010001111 01
111100010100110101111011 10
010001101101001101010111 00
111011000010110001001011 01
101110001100010100101011 11
110100000001011100100101 10
001101001010111001110010
111010110111111110000100 0
011000001100110110001010 00
101011100010001101010000 1
000101000111010001100111 0
000000110111010001111111 011
111010000000101111100110 10
001010011100001010010111 0
100010000001010001000111 0
100011010101001011010100 1
010110001000111010110011
n = 25

n = 26

10111011100110011101001000
01010010110101101101100000
10101000000110100101100101
11101000011001000111010010 0
01001010000100011101110000
00010101110001110000010 1011
11000011010101000010000 1100
00001101000000001011010111
01110111000011011111011111
11000101111000100100001 1001
10101100110011101111100 1110
10110101010110001001110 110
01111110010000100010111001 1
10111011111011011010000 0
10100111011000110010010000 0
00001011001110010000101000 1
10110111010010001001011001 1
01110010011010100111000 1010
10111011100010111011011 1011
10001011110111010001011 1010
n = 27

01011001000010101001011001 01
00100100110011110010101011 11
10011000011001001000001001 00
11000000001011100101101011 11
01000011101110001010111101 10
01101010111101011000101001 01
01001111000001111000111100 0
01000110010100100001010110 10
01100000010010100110101010 00
11111110101011101011101011 01
10000010111101000111110100 10
01001010010000101001011111 001
10111100000000001110010100 1
00010010111100001110011100 00
10000111011111001111010100 00
00111000100110010011101111 0
01101000100111010010010110 00
01100011000101011110101001 10
01100101001001110011101111 01
00010000011000010111010100 1
n = 28

10111001011110011010010001 100
10010011000111111010011100 00
01100000110011010010110101 001
11101111000111010011101011 101
00110000011101111001110101 001
01000101100001111101011 100000
11010100110011011100110011 000
01000111101111001101001100 11
00111101011000101100111101 110
10010011001100111011100111 01
00011000010000101111100000 000
00101110000111100110000 110100
00111001100010111100000 100011
00101101000100011101010 101000
10001010001000001010101 000001
11111111010100000000101 11001
01110100001110001011110001 001
01000100111100100000000 010100
10001110101100011000100 011110
11101010110010100011110000 010
n = 29

10100110100000011100001 0011001
01110110000011010111110 1010010
10000001110110100001010 0000000
10000001000110101111001 0111000
00001010000001101100001 1170010
10110110111000100000111 0111100
10100100001111001011100 0000010
11111110000110010000011 0110010
01000100101111001100110 1110010
01010110011101101001000 1010100
00001111001111001001110 1001010 1
10011000110010100110111 0110110
00110110011100001010100 0101000
00000000010010110100011 1100110
01011000011110011010111 0000110
10111100100130001100010 0111110
11000010001110100100001 0010011
10110000100111010101100 0011100
11111100101110000001010 1110000
11111011101001110100011 1100110
n = 30

10000010010111000100110 1010001000
00101011001100011101101 1001001
00011001010100101001100 1101001 0
11001110010101001100000 0011001 0
00001010000000100001101 1100101
01001100010110110100110 0010010 1
11110010010111011011000 1101011
00001010110110001110111 1000011 10
00101110110100111111101 1101110
01000000011111000100101 1010000 0
10101001010101000010101 1000110
01001110011000001111000 0110110 0
10111101111111011110001 1111000
01110010110100011000001 1010110 1
00000110111100111001010 1010 11
11101111010110001001101 0010011 01
10110001000100101000100 0110100 0
10101110001100110011000 0111001 0
00100011001101100110101 0000010 0
11011111001011111010010 0000000
n = 31

10110000001000001110010010 011000
10000101010011111100110101 101110
00111010101110000100100101 101001
11000100011100001000001 0101000 01
01111010110110111000001 1100110 01
10111011100001111000101 0001011 10
11000111110000011001100 0010001 1
00010001011001101101000 0101010 1
11111011000010010001000 0010010 00
01011100110100001000101 0110011
00010000111101000101011 1000011 0
01001010010111001001011 0101001 1
10101110001010110001111 0010111 00
01110111010011001100100 0011111
01000100011100001000101 1101100 11

01111011110010110010111111010101
01000100010000100110110110000110
01110010001111100001100111000000
10011100001010010100101000100111
01110111000011101111111000001001
n = 32

111100000001000100000101101011111
001010101110110101011001100010001
100010011111001011101111011000010
010100110000001000000001100111010
001011101101011011011000011011001
001001101111011111000001100011111
001011101111010000100011000001001
110011000110110101011100101010010
111000111110001000001011011110011
111100011110000100110101001111001
100111000000010010111100011010110
001101011110100000100101101100100
100011011101010101010101000011010
000011100100010100110001001101000
010100111100101011011010101010000
100011110110000101111010010110100
011001110100100101000001100010001
001100011101110110100111111101110
101100100110110111000011011010111
111101111110110110100000000000101
n = 33

0110000100101000011100100010101000
0110100000110111010110000101111001
0110000111001011111010100001011111
0111101001010100000011101100010111
0110111100110101101001100100001001
1111110000010111110100001111101111
1000100101001110001110100100111010
0011001111111010000100010111101110
0001110000011010101010011101011101
0000101101010010110100001101100110
1010101011111011111001100101010100 0
0010000000110010001101111111000001
0101100000010011100011000011010100
1010110110011111110111101100001001
0000111110111110011101110011011111
0000100101110101011011111110101001
1111001010011001010000100100100100
1001010000101001101110100111101101
0100010110111100001101111100011111
0110011000000011001101011100010011
n = 34

00101100001111000011010000100110111
00000111011001001001011111000010110
01001110101001100101101101110011011
11000001011111101010111101110101010
11111011010010111101101110011101110
10100100100110110110011000010011001
11100000110010111001010101110000110
10111001111001100001110000010010110
10110010011101110001101001011010000
01001011010101000111101000101001111
00010111000010100011000001111100001
11101100111001011111110000110011001
00011101100010000000100111001011010
00001111111100011111110000001101000
11100010011101110001011010100001000
01010101111011001110010101100101000
00000111010110101101000001000011110
01101100011101101101001101101101010
10100100110101100011001000101110011
10110110001110101001111100011110011
n = 35

110011010110011100111011010011100011
101010001000001110100001110111100011
110101001001000000000100110000100011
101111110001101110111010110011001001 0
000011010111100110001011101110000010
101111010101000011111000110001100000
111001110010000001110000001100101111
011100001110010011011101110100101011
110110101100110101001011000011110100
010101110100101000101011010010100100
001100001101111000101100000100111011
000011101101001110101011011100001110
110010111110000100111011011111000001
000000110010000010010111001101011010
010110111001111010000101100111001100
110011100110101001101100111011010110
011010101110010100011100100001001101
100000011010101101000101101110010111
111111011111101000110001100110100110
000011000110001000011011000001111010
n = 36

1110101110111001000110000000111111111
0110011001111011000100000001111010100
0000011101001000110001110010110000000
1010011001000110100010110000000100011
1111010011010000011011111001000000111
1100111001111111100011001111001001101
0001000101100001011100101001100000111
1111011100001100000001111100011011100
0011001001011010010100000011000011110
1100001101001110000000011101000100001
1100000100011001000000000001110111011
0000111110111010100111010010101110110
1111100000101000001010000111001000100
1001011101010100001001100000000011010
1111010011010010011000000001010011110
0111010011010011101000111100000000111
1100100110001011111001101100100011101
0101101000000111110111011110000010010
1111010110010000111000001011100001100
0111000011010011111010100111100011011
n = 37

00110011101010101001000110110101101001
01011001010000100011010110010000111101
00001001010111011100101101011000110110
11111100001010100001110001110010100011
10000111011110001001000001000101010000
11001100011010000101101111111111101000
10000010011011100000001101110000001100
10010011111000111011100011111100100001
10000101011110111110100001000000011101
11110100110110110011111110001001001001

101000110000011000110011101010011101 10
001101001010010111011010011011010 10
000110001001100010001000000010100101 11
111111000100010110001000000111110101 10
010010000010010000010000110010011011 01
100000010110000001110110000001000 100
010011010101111100011111010011000111 10
001001110100011110000011010110111000 00
111000110101010000111000111000000000 11
011101010111110000101001011010000100 11
n = 38

000011011010000000000000010011100011 11
110011111010100101010001100101000011 1
111010001100101000011110000101011111 000
100100101010101100111111010001000001 00
110110011101110100000100111100111111 01
011110011111101110100010100111111001 110
101011001000110011111110010011101001 001
100000000111111101010000101000001110 11
101101000000100110101001110110101111 000
111101101101010111101011110000000111 0
100001111000101011001010011001010110 001
011000010101000011010000101010010101 00 11
110001000001010000100110011111001100 010
111001100100000101010011000011111110 010
000010111010111010000000000001110111 0
010000010111011100000010000100000111 0
n = 39

001000001010000011011100010010000000 1010
100011100011111101010101100000111101 010
010000011001111110011101101111111011 01
110101001100100101011101000001100110 1001
101111000101101000001000111000101111 000
111110001111001001100100111100100010 0100
111011000010101100011010101001000001 101
001110010101010000001000111000111111 0010
100110110100111011101110010010101100 101
010000100101111001001001000000111110 1
100100001011100001001100111100110000 111
101010000000001111011101110010111000 1001
000011100110111001111011101101000 1000
110010011011111010111010010101 0111
110010000101111010111000110100010100 111
000001000100000010110010101100110100 100
n = 40

011000101000110001110001011111001000011 0
101100001010000000010011100100001111 100
100010010110001101101010011011101101 0001
011001011110011111001011111100011110 110
000101110010011100011100010101010011 111
001000001100001100100101110011010111 101
000000011010011111001000001001110011 10 10
101001000111111010011101100011101011 110
001101101101111001010010001110010101 1100
101110111010000011110010011110111011 101
011100011011010000101100100111010111 1011
101010001110011010010011101000010011 10 101
101100000101011000011110110000001 1000100
011100100001111000000010010010000101 000010
101111011110001111011110101101101110 1000
010011011111100111010011110001000101 0100
110000010010110110010111011011101000 11 011
110110011111111011100010001011000 1101001
111111100101011111100100011101100 10 10 010
001101101110001110100101000000101 1000011
n = 41

110111101111000010000010011110001100001 101
101001100000111101011110011110001000001 111
100111101111001101011111100110111000111001
011100010011010101100010010010110001 1101
011011111000111011010010101100011101010101
111111011010101111000000001000100100000011
100110011000010000011011011010010000011101
001100100010101011011010011010000111101
111000100000111000111011010010100010101010
011100011000010000100001010011010010010 11
101000000111001100011010010010010100101011
011100110001100100010000100111001011010 00
100000001111010110111010011000111010111 111
110110100100111011011100100110011011100 101
010000000110000100100101011011000111010 0010
010100101101110111000111010100111011000 1111
001010011000000101111010110011010110100100
011000011001011111111101001101001111001
n = 42

001010100010101110001111001110010110101010000
011001101010100010110000001010110010000 1110
001011101101011101010111011000111101000 0010
000100011001011000011101001101101111 10011
101110010010111011101111001001001110111111
011010111101111000010001011100010011101111
001000101110100011111011100100100100000010
101111100001011101000101011010100100 1000
100110101010100101100101111100100111111111
001110000001000000110111011000111111101000
110101100101011001100011011000001011101100
000011101011010110111010100011110101101 10
010000000110000100100101011011000111010 0000
101011111000111101110001111101111011101101
011000011100000100000100111001011010001 1010
100100101101010101000101100001111101100000 1
011010000010001110011001000010101101010110010
010100101101111011100111010100111011000111
011000111001111111111010011 0 1001111000101 01
100011111001001011001010011101111010000 1
n = 43

110001011011011010011100000010000100110111 11
001011000010001101101110101111000001011000011
111010001010100100100100110111010000000 11100
011001010000010110001101000111100111010000 1
001010001111110101000100101101110100 1010 1010

1011110010110100011100111000111100000111100
0010100101100001011101000001101101101100101010
0000001101000100011110101110011011011011100
1101101101100011001000001110100000101101011
1111011010000111000010011100011011100111101
0011101010011011011010111011101010011110101
1100101110111101001001101100010011011011111
0011010110100001111000011111001100010100010
0100011010100011000100001010100000100001110
0011001000011100100111000000010110011011010
1100000111011110101010111010101110111001101
1011111010111001100000001111010000100100111010
0000101110010101011001110011001010100011110
1110000010010000111000111111001100010100100
1011111010111001100000001111010000100100111010
0000101110010101011001110011001010100011110
1110000010010000111000111111001100010100100
0101101111000011001100100001011111000100010001
0010001101111010100110011100011110111011011100
0000101110101100101001100011111001100111111100
1001111011000101010110111101011110101011011101
0011000101110001001101010101100110010111010010
0000100001100011111110011110100001011100100010
0000101010101011000101010111111101010100000001
0000101000111010001101011110001011110001011111
0000100001010100111111101001000110111001101110
1000101000010110010010011100100110001111100100
0100100001100111001101010111101111111111111111
0110101111010110100000001100101111011011011
1000011000111111010100010011010110100011111110
n = 45

1100111100100100011000001001000111011110000011
0001111001001000110000011001001000101001010110110
1011000110100010010000001001001100000000110011
0000110010001111111100001010111100010011000010
1101101101011101101100001101100100101001010010
0000011111111010011010101011110010100101100011 0
1010110101101011010110101011000000011101100011
1101000101110010010011100011000010100101100100
1001110101001101011111000111101110110111101101
0111111110110110100010011101011110010000000010
1000011011000101001000101101001011001101101011
1100111001010101100101001111101001001111000101
0110000101101111010010111010101101001001001110
1110000001001100101011110001110111010110110011
1011110101110110111110101000001000000011100111
0111001001100110011010000101101111100100000011
0011000111100011000100101001101011010000011110
0011101100010111011010110110010111010010000010
1000101100010110100010110011101101011101010100
n = 46

0101000011000111010010100110011101100010000 1011
0001011000000000110011111110001100111011010 1010
1101101011001000101111111000111011101000011 1010
0111001001110011001001111011010000101101001 11110
0000101010101111011000011110101100101010011 10110
1111010111010101101010001101110000011000110 0100
0110011100000110000100010110110000011110001 10110
1011011010110010000100100111010010011100011 1100
0010101001111101000000110100010001010101010 0111
1001111001110101010110000111101010011111000 1011
1011111001000001010011010001100010000000011 1011
0011100101101110001100000011100001011101101 0001
0101101000110101111011001000100000101101101 0101
1000110111000100101110101000110111010011101 01010
0110100111001010111101101300110000110111111 1001
0000011101101001000000101001100100111010101 10011
1111010011010011111000110101110001010100101 01100
0000100000001111110000001011100001000000010 01010
1000110001010001000100100110101011110011001 01101
1001010101100000000011001111111000010001 101010
n = 48

1000010000111001100111001000001000000111011010100
0001000101001110000101111011111011011111001101
0011110010011111011001001110100001101100001000001
0111011100010010010010101011110001011001101 1011
1100100000100111010101001101011011001111111001 10
1010011101100101010011011010000010001011000101100
1100111011111100010110100000111000111011001010000
0111000111101100010000001011010111000011111111 1
0111101011101101000101100000101100001010111011110
0010001110010111111000110101100111100101110101100
1011000010110011010011001101001011100000010011 0
1111101010110000000011001100011001101010011110011
0101111101001111001110000101100001000000000101010
1100101111000010010011001100000000100010100111101
0010100000011011001010011010101010111011010101
1110100001010100011011001011010111100100101 10110
1001010000111010000110011010011000111001110011101
0100000001101101111000111111011100000000011101110
1011000100001010011111101111000011010110011101 00
0010101001110110100101101101110110110010010000 11
n = 49

1000001001010000001111111001001010000100001111010
0010101010110010110011101001010011011000001010100
0110100110011010101100010100010001110111000001111 1
1001001101010011001101000010011010110111011110 1
0111101010100011100100001010011111101111010 11010
0010110011101000001110111011000000101101111010001
0100100001101001110111001010110001001001100011110
1000100100101101001111010010101011000100011101111
1110100100001000111011110101011111000011101011000
0000110111010000101111010110011010010000001010
1000101110011100100011101111001111001110011000100
0110100011111000101011001001001110101110011010101
1100010011010101110111111111011111000011000010000
0010101110101011001010111010110000101010001010001
1111011101010011011110000000110100000101010011010
0010100100111010000110011010011000111001110011101
1001010000111010000110011010011000111001110011101
0100000001101101111000111111011100000000011101110
1011000100001010011111101111000011010110011101 00
0010101001110110100101101101110110110010010000 11
n = 50

1000100000000100010110000000011100100111110111110111
001011000000111100000011010111101001100100110010010
100110111101111000101000110111000001101111110001110
111010111110100110111010101000101011010000001101011
101111000101000111110110101001100101110101001011111
001010110000001101101011101000111010011101111110001
010111000000101110100101000110110110011100000101000
010111101111101101010001111110000011101111101110100
1100000110111110100111101011001000100011100010010101
100001101010001100001101010000000100100101111010011 1
101010001101101111111100011011011011000010010010001
100011101111100101001110110010110111111010101101010
110011101010111010000011010000001010000000100111100 11
110001100001101010001101000110100000110001011010000
100001100110010100000101010111000011111110111000000
011011011101010001011101101111101010101001100100100
0110010011101100000001110100101001100100111101101101
010100110101110110011011110101010100001111011101110
10001010101011111100000010000000011011000111001100111
1101111111001011101101110001111010000111110011100000
n = 51

10011101000101101111111100110111110000101111100111111
1100010111001000000110101000010000101100100000000000
0000010000100001111001110000100011000010100000111100
0100101010101001111101010001100000110001000011001010
0000010111100001000010010110011010011000001110000110
0111010000000100000011000000011100110101000111010111
0000101101010100110100100101101101011001111011110111
0110011010011111101000100010001101010011010011001001
1001101001111011001010011010000000001100000111111001
1110000101100010011101011000101011010001110111100110
1111101001101011000101001111101010101001101010111101
00110001011000100001011100000000101001110110010100101
1011010011001000110011000101100100110101010101110001
0101101010010011000100010010100010011001100000111111
00111101000000000011011111110001000010111001110001110
0010001011010110001001000011000101110110111010110010
0010010100101011001000010011010110010011000000011110
1100100010101110011010110010111101011001100101011000
00011000011111110000001110110101101110000111010101001
0010000101111010100110111000000111000010000010101000
n = 52

11111110001101100011010001010001000001110011110100000
0010010010101010000000110100000001001101100001111101011
11100001111111111110011000001100010000111001000001010
01111100101001111000111011111100111100001001101100010
10100001110111011000101011001100100110100111100100000
00110001101111101111111110100000111000001101111111100
01011000011110001011101000001000010010101100000010100
10011001101001001100011011010101110001111100001101000
00011101011001100011001101001100010001011111001100101
11100010101000110011101110110110000110001110001110010
11101001110111000111000010111111111010110100100010110
00010101110110110011101000010010000001111100101101001
01101010101001001111100000000111001111100111101111010
01001011110101101001101001011111111010111000110111001
11101000110111010100010100100000001101100101101110100
0110000000101000000101110000101110000011011100110000101
011110100000010010001101101000111111000100001110011111
111000001001000100010011011011011101100111110000100101
011011001110111110111101001110001100100000001001000101
01110110001100110101011100101001101011010111010001100
n = 53

0010100010011100001001010100111000101101111111111101110
1100000110111000000000101010110010000011000011101011100
110101001000001100100101111110000000111001100101001010
001011100110011010000101110110100010100101100100011111
011111011000100110100010100001111101101111000100110011
111101110110001001010101001110001110011101010100101001
111111000010110011001111100000010100010101100010001000
1011000110010110000001100111111000000010101011000110100
1001111110001000110110010011010010011111011000100100101
0101011101101100011100001000111010010111000101111111011
1110001000010000001110100001110010000111011010111110001
01110010000001011110010010100110001011110000001111010011
111100010111011011100100100010010010110010011110101011
1100111000000010011010100001000000110101100111100010000
0001010100101111011111000111000111110111010010101111111
111011110100000011000010011100100011100001010101000001
001000101110010101010111100100110111000100110010000111
101010000010000011001111100010000111110010101001000111
011101100101101110101100001110101100100100001011100101
000000000111100001101000011100101110001011001100111101
n = 54

01111011000101010110101001001010100011111100111011111111
0010110011101101000100100010100100100110011100000101111
01111100000001010100011111010011001101000011001011001111
0111100101000110010110111011011001100001110111011001011
11001110010101010011110010101011011100000011110001110110
00001100000001110110101100110101111110000001101100110010
10111110001100111111000100111010110010100110101000000000
0010010011010111101100110110100111101001110100010110111
0100000011001101110111100010000111101000000100001100010
0011110100000011000100101100111110010101000111000011100
1001110000110000001000011110010101110101110111111011100
10111110100110101100101100011100110101110000101010000000
0000101001111001100110001100101000000110011101101001010
11011010111000100001101110010100000001011010111010000011
1000001001111111001010110011100110000101010011010011000
1000000010011010100010101111011001001001110111010101010
0110100101100101111010111100110100001010011110010011110
0111010100010111001011010010010101110101001000100101001
1101001111101010100011100010010011010010001011101110100
1111110001011001110001000000101100011011110111000011101
n = 55

10100000000110010001111010100010110010111000110110001101
000011101001100100010000000001100001110101110101110000100
00101101011101010001110101000100110001001111111000011010
00010011010111001111100001001011001100111110011100001110
10110010011101011100111001111100100111010111101100001110
01000001111101000110001100101001100001001010100001000111
100000111011011000000001000010110001001000011001100001011
00001111101010010100001101011101010100110010001010100111
01011100011111001001001000110100110110010100100011100011
1111001101001100110101000111111100000000010111000000010000
111111001101110100110000000110011010000000011100101010011
11000010111110111100011100110010111100110001010101110101
110100110111111000000001000111101001001101001111000010111 0
00000100010101111111101010000010010011100010110010000001
00000111000100111010100010110101100110010100000111001100
01010011011010001000110110110110001001001110100001100001
00010011000010001111111111101111001110001001100011110101

1101110110101011101100100000110111111010100011101011000
1101000111001010111011010000010010100001111010001101011
0010110100101010001001011101000101000110101001011000100
n = 56

0110011100010100000010000001010000111110111111110001010 01
1000101110110101001101101111010011101111001111010010100 1
0001110101011110001110010100110101101110010000001010111
1001000111100000110011001011010101101101000001111101010
0010000010001000100101111001000111000100101001001001010
1100100000001110101100000110111110100001010011010000011 10
0111100100101000101011111110010110111111111001010110100
0000001000100011100110010101010011000001001010011111101 10
0100011110011011111101100100100111001010111011001001011 00
0101001000100100110010100011011010010010010111111001010
1010001001000100100110111001000101100111100100100111010 111
1100100000011101011000011011111010000101011111010001110 111
0111100100101000101011111110010110111111111001010110100
0000001000100011100110010101010011000001001010011111101 10
0100111110011011111101100100100111001010111011001001011 00
1000000101111101101100011110000101100101111100000101100 0
1110100001010010000111100101000010100111100010001110011
0000111011000000100110110111001101010011111100001111001
1111101001100100100111001111100111010110100101000110100 1
n = 57

0000110110100100111101110111011001111011001110010010100
0100010010101000000101110010001011000110011100011010000 01
0111101110100010010011001110010001011000100100111000010 1111
1110110010100010110110000100100010110011110001100001110 111
0011001011010101000101011001010000011001001110010010101 1101
1010100001010011101000001001001010111011000100111000011001
1111000100110001011011011001010010111011000100111010011 0100
0100110000110011110100010100111111111010001000011001100 10
1110101101110011010011010100001001011110100001100111010 0110
1101011010001110111011101010000010010111100100101011011
1010110000101001110100000100100101011101100010011100001 1001
1111000100110001011011011001010010111011000100111010011 0100
1000000011001111010001010011111111010001000011001100 10
1110101101110011010011010100001001011110100001100111010 0110
1101011010001101011101010100000100101111001001010111 0010
0110000011100101011000111110111011100100010011010111 0100
0001011010111101101010100101010110000111001101100000010011
1110011100101100000001101001101000010100010110010100011010
1001010111001000111001001111010000110011111011111101011011
1111111010111000000111010110010110011010001010100001011111
0001110110101111100001010111110011001000100011111010011110
n = 58

0110101100100011110010000001100111010111010111110011101 01
1100010100000101111111110001000101011111100110010111011
1100000011011011010001010101010011100011100101011001 10
0011010111010101110110100100001111111110010110100001010 00
1010110000101001100101100110101011110000101101100111101 10
1100000110000101111100111001101101110010100011110100110 00
0101110011001000101010001100111101100011100100110101001 10
1000111100001101101101010110010001111001111000101000011 1
0110111100111111011011000101110001000001011010011101010 11
0110011001111110011001000101110010001100111011011010000 1100
0100100100110011101010011111010000010010010010001001100
0000000010011111111101010111010100110100100001010101 00
0000001100100010110110101010111000100001110111011000111
0001110011011010101001001100011100010010000000010110111
1100101110110000111001100111101111001000011000101111011 11
0100110011011010110110101010101101010010010001000111101
1001001001001101010000111111010110101111001101001100001 1000
1101000001110111111000110000111001110001001100101110011 0000
1000011001110011110000110110110010011101110001011001000 110
0011111001010010101111011111100100010001000101100010000 1
0110010111111001000101111010111011010100011000101101100 111
1111101111111111000010010011011010000010001100101001111 010
1011111011001010001101111111010101010110101011101101011
1001001101111110011010100010100010010010011000010001 10
0111001011010001101111010011101100100111011000101101111001 111
1100100000100001011001010110100101110101001101101001111011 10
0000010111001010100000010111101111011101010110011000100
1010100111110100011111110011010001010001100011001000011
0101001011110000000100101001101001110000011011001000011 00011
n = 60

1111101111001011011111101000111101010000010101001110010101011 1
0101100000110000000001010001101010100110001101100000011011
1111010000001000000111010011100111000101110111111011000101 1
1010111001100110100001000000001100000010001111101011111110001
0101000101010000101111111010110001001010111011000000101111110
0111010110100100110110001011100000100110100010111000010111110
0111101100111011011000111111001111010101011011100100100000
1000100001110100111101100011111100111011010000100110101010 1
1100110010000110011010110111000111111001111010100010011100111
0110011110010100101100001010111010101010110100100100111100111
1110101000100100110100111000010100100100100010110100011101110
1001011010000010110000111011001011010111001001010111100011 0
1000111000110000111011010000110110010011010010010010001 1110
1000101110110001000111011100100100001010011010111001010000101
0111011111111111000110111011010011000001010001100111010010111
1010101000111010010110111010001111010011000010100111000111011
1100100000100000101100101010000100100100100100100100100 1100
0101011100100111010001100000010101010001001100101101101010110
1110111010010001010100000010101101010011010001010010111110
0100101101010101010101000000010101101011010001010010111110
0101010111001001110100110000001011010000100110010110110101010110
1111011101001000110101100010000100111000001111001100001111000
0010010110101000100000000001011010011010001001010011101101
1100010110111011101110010110111010101100100001100100000101
n = 61

1000110001011101100011111010100110010010110000110011000111110
1000101110110001000111011100100100001010011010111001010000101
0111011111111111000110111011010011000001010001100111010010111
1010101000111010010110111010001111010011000010100111000111011
1111110011101110001010111111010000011010010101010101110000 1
0011011100001011010010101110001101010000111000011000010010 0
1010110010110110001010110001010101100000111000110000000010111
0111110111100101111111001010101000111000110000000010000 1011
0100010010001111100011110000010101100101011111000000100011011
1000000100110000111011010000110110010111000101100000001010001
0101000000101111111010000001100110111010000000100011110110 0
1101001111010111000001100000011100111010011011000010100 1101
n = 62

000010010111101101011011100100110000101110000101001011000110
000010100001111000111011110011010101100110100111001101010100
100100010100100101100101000010010000100001100011011010011000000
100101110010101010010101100000011000101000111011000100100000001
101110001110010110011111011000000100110010110111010011100011110
100000001000100010111011001101111011001111101010011000111000
011001101111011111000100111101100010000101011000100101001100
110010001101110001100100011000100101010011101100010101010110110
n = 62

110100110111011011111111011101110010011101101010111011100001111 10
000100101110001010111001101001100001000001011111011100010101011
000101100000110100100100100000101001011011111000100011101100111
101111000111101110011001010001101110010001110000100010000010110
110000000110000111101000111011011011110011100010000011111011
001100101111010000101010000011000111011010010010001110011010101
101100111101010000101111111100000011010110111101000001011100001
111101111001011101000110101001011001101010110100010110101100
100100011100010110111010100011011010110100100110100000001011001
010111110100010100000001000010101111100111111111001111000011000
101010111111011111000011001001011100111000111111001000000100101100110100
000111001011000111001100100001011010010001010011110100000100001
011110000100100111000101000111111110110010000001001010011000000
001000100000001110011001000010110100100010100111010000010000011
001001011011110010010011001110111001000010011011010101100011110
001100111011111000011111111000000100101111101101011100111000000
001001011011100100100110011100101011010010001100100100000111000 1
n = 62

110100110111011011111111011101110010011101101010111011100001111 10
000100101110001010111001101001100001000001011111011100010101011
000101100000110100100100100000101001011011111000100011101100111
101111000111101110011001010001101110010001110000100010000010110
110000000110000111101000111011011011110011100010000011111011
001100101111010000101010000011000111011010010010001110011010101
101100111101010000101111111100000011010110111101000001011100001
111101111001011101000110101001011001101010110100010110101100
100100011100010110111010100011011010110100100110100000001011001
010111110100010100000001000010101111100111111111001111000011000
101010111111011111000011001001011100111000111111001000000100101100110100
000111001011000111001100100001011010010001010011110100000100001
011110000100100111000101000111111110110010000001001010011000000
001000100000001110011001000010110100100010100111010000010000011
001001011011100100100110011100101011010010001100100100000111000 1
n = 63

111010101110100110111000010000011000110100111000101100000000010
010011011011001100111010110100110101011010110001010101011110111
111111011001011111011010101011110110001001101111011001011001110110
011011001011101010011101011010000011001100110011001101011000111
011110101001100001101111110010101111100000100101111111010111110
010001110101011000111001100100001011010010001010011110100000100001
010111000010010011100010100011111111011001000001001010011000000
001000100000001110011001000010110100100010100111010000010000011
001001011011100100100110011100101011010010001100100100000111000 1
n = 64

Solução 103600_extra.h

0100111100
1100100111
1001011111
0111100110
0010111000
1111101111
0100111101
0010110111
1010110000
1100100000
0110000100
1001011101
0000000011
1111111011
1010110111
1111000100
1000011101
0100010001
0010100111
0101101110
n = 10

00001111110
01110001111
10001000101
01001011010
11111010000
11010011011
00101011010
00110101000
11110100101
01110101010
10110001101
10010111110
10000111111
10000111011
11001010111
00010010110
00100111011
00111101111
00000100010
01011010001
n = 11

110011100000
011110010100
010001100101
101010000000
101111111000
011010011010
001101010100
110111010101
111011011111
100001101000
000111001100
111110100001
010101111101
011010111001
110111111000
011110110100
111101001011
111001100010
110010011110
101000111111
n = 12

110000010001
111011000110
111100011110
101101110101
101001101001
110010100000
011011000001
100111010001
100011001011
101000101100
110111001100
100100100001
110000010111
000010110010
101011011001
110001100110
100111010010
001111110000
100011000111
010000111010
n = 13

001011001100 01
011110110001 01
100010011110 10
100111011010 01
111011100100 01
110010000010 01
110100100110 01
100111010110 01
000111100000
100111010010
001011100000 00
100110100100 1
011010010001 00
011001011011 00
011101100010 10
n = 14

00100101111 1010
00111001011 1100
11000000001 1111
01011001110 0000
01111111011 1011
11101010110 1010

011010110001111
000101011110001
010100010111010
001000110010011
011110001010101
111101110010011
001011110000111
001010011001010
111110001000010
000111101101000
101011000110110
011111010111011
001011110000010
001000110011101
n = 15

101100010110100 1
111110010001100 0
101000010111000 0
101101000000011 0
101111100001011 1
010011001110000 0
110001000111111
010100000010111
110100010110011
100111000010011 0
111101000010110 0
100110100100001
010000000010011 0
100001110010111
110011100110111 0
110001110000111
101110010110011
011010011100011
010100010111010 1
111110010101101
010100000011010
110111000010111
101110000101100
010100000010111 0
100111100010010
101010011100100
011011000110011
110001001000011
101010011100000
101010100111100
n = 16

000111101011000 10
011001000100001 00
001111111000011 11
100010011111001 00
000101110101110 00
111100011111001 10
010100101011010 01
011010101101000 11
010100101011101 01
111111100111010 01
001100011111101 00
000100010110011 10
101010110101110 10
101010110110110 00
000001110000101 11
001000000111000 100
111000100000101 101
111110000001110 10
101110000101111 001
111101100100100 101
000010111010101 111
001100101100011 001
100100100000010 111
110001001000001 011
n = 18

011001101010000 0010
010111101100000 1110
101111001011111 11001
010111001111111 11000
011000000001000 111
011101101011010 0000
101001101010100 1111
001001100111110 0001
111101111010111 01010
000010010110000 1011
111001101111110 0000
000101010111100 1101
011110011001000 0010
010100000110011 1101
001110110100110 0110
101101110010101 11100
110010110100001 0011
001000000001111 00011
001000011011010 1010
100011111101011 1100
n = 19

101101111011110 0001 01
100010000100000 0100 0
100011100001100 1110 0
110011101100110 1100 1
101000001101010 0010
111100010101100 1010 1
111011011110110 1001
111000010110101 1010
011101100001011 0100
101001110111111 1001
111010011100011 0000 101
000110110110001 0101 010
000011111010100 1010 100
111111100010100 0011 0
000111110011101 1011
010100000010100 1010 0
000101111110011 0010 0
000000001111101 0111 1
011111000100111 10010
011010011011011 01110
n = 20

001001111110001 0111 11

100111001011000110111
101010001110100100010
011011010111001011001
101000110001001101100
110001010111010001100
010101110011110101001
101101110001001001101
110011100000111010101
110011010000000101111
000001110111010001110
100101000001100001110
011011100000011000100
111110000100010110101
110100000001001011011
110100011100111111111
011100111101000010100
000010011101010100111
000101111000101000101
100100001000110000000
n = 21

101101010110000011110 0
100000000100100100001 1
111001010000010011001 0
110101010101010100011 1
111100010110111001001 1
001000111110111000001 1
001000010000111011111 1
011110001100011110000
001011101010001010111
010110010010011111100 1
011001011011101010001 0
101100110111000101010
000100100100110101110 0
000110010111110011110 0
110100101010111100111 0
110000111111100110000 1 0
011100011001010111000 0
101111111011001110000 1
100100100000001110000 0
010101100011000001011 0
n = 22

100011010110000010000 11
000001001011010100011 00
011100010010110111111
010100010110101100110 00
000101100000000100011 01
100000100111001000010 00
100011000110110010000 01
011001110111110010001
011011101110000110111 001
011011101011111001101
001010011010001100000 00
011011100010001111100 1
001111001000101110101 01
110110111010100111110 1
110100000000011001110 11
100001000111000000011 1
011111100111011010101 1
110100011111111111110 1
101101010111111010000
n = 23

000110111010101110100 100
000011100101111000111 00
111111101110011111111 000
110110111011110000000 111
111101011010011100110 11
111011111010110001101 01
110101011111001000010 00
000110011000100010110 001
100010001111001011010 01
000110010111001000101 000
110111011100100011100 100
011001101000001000101 001
111010001000011001111 11
001011111100010011011 001
011010111001110100110 010
001111010010001100010 100
010011001010111010110 100
000111010110111001101 100
110010101001011111011 11
001001110101110101011 010
n = 24

001000011000000110010 0111
010110000000001100010 0011
010100110011000110000 0011
011100010101100110011 0101
111110100000101100111 1101
001110001100010000011 1111
001100011100111000110 0011
011100101011100010101 0010
101011000010000110001 0011
011110101011100100010 0100
000000000101011101111 00
111000100010101001110 1000
101101001111110001100 0100
011001001110101011001 110
100101100011101001011 0101
010010101101011100101 0011
100000001010001101000 0100
110001010001000110001 101
000000110101101000100 011
011000100000100000000 1111
n = 25

001110011011010010110 01110
000100100111110110101 10000
101101011000011000100 1000
010100100000111110010 01100
000001001111000111011 11110
110000101110101111110 10111
111011000011010000101 01110
101000000010011010101 01010
110010111001100010000 11000
010000101111001001111 01111
110110001010111001000 10010
101111111000110111000 10011
111101110101110001000 01101
010101101110110110111 00001
100011111101010001010 11000
100001110011001000010 01010
110010001000011100110 11110
010000010111010101001 10100

0111011001010111001101011 0
1110011100100100100000000
n = 26

1011101010111001001001111 10
1100100011101101111111000
1101010110011001100000011 00
1000110000001111111011001
0010101111011100100010101 0
1010111101101011100111011 00
0001010011100010101111001 10
0011010101001001101100000 0
1010110110101000000100101 0
0000000100010101001010001
0000100001110100111100010 00
0011010011110100100011000 10
0000010011101000101001 1000
0111001111000100101100000 11
0010100001100101101111010 10
0011011001111000100111101 10
0001101000001110001110011 00
1001011110010010001001111
1011111100101110110011010 10
0100000111001001110001110 10
n = 27

1010011101001100000100011 010
1001101110110000001010111 001
0111001000111110001111011 110
0000100001111010011101111 1
1100010110011110000010011 010
1111100101100000100100100 100
0011001110001100101001001 00
1111110110101110100111100 10
1011000001000110010111100 010
1111011111010011011010000 10
0000000111110101100000100 11
0100110011000001001000011 0
1010101110001100110000011 10
0111000001001101101100110 0
1100100011001111010110101
0001001110100101001011100 010
0000000001110101100011000 10
0001011011000001001011000 01
1101011010100001100000100 100
0100111010010100110011001 011
n = 28

0011101010001010001111111 001
0100101100110100011101100 1010
1100100011111100010111110 0001
0010110111000010000001110 0011
0111111101001011100000110 110
0100011101111001100011110 0101
1011001010010100100000010 1101
0110101101011100110011100 111
1110000100001000000011001 1000
1111010100100011100000100 1110
1100001000100010100110000 011
1111111010101110110101011 1011
0000001111001100100011010 0101
1011101001101011011111000 1000
0101011100110101110001011 1011
0110000101101000011001 101001
1110111010001100010100000 110
1100111111011101100111 01101
0110110110111000000001001 1011
1110100101000110110101 100011
n = 29

1011010011100010110111001 01011
1101101110111001100111111 00011
1011010000010010011011111 00110
0010111010111000000110000 0101
1111100101100000101111010 01000
1011110000011010101111011 01000
0110010011001010101100000 0000
1101110100100010111111100 10111
0011101111010110100001010 110
0001100110001011110111111 10111
1101011010100100000110001 1011
1111000000011101000100101 01001
1010010000111010100000011 00111
1111010100011011001001001 0011
1010110111111000110011001 11000
1111011010111001100010010 10010
0111000100111111010010010 01111
0010010111100110001100001 11101
1110101010000001000110100 10010
1000101001110010110011000 01000
n = 30

1111001001101000000001111 111011
0110011000101001000110011 111001
1001010010110101100001101 000000
1000100010110001110111101 101101
1101100111001010010001000 000010
1011111001011101100101100 001111
0110101110010010000011100 010001
1001101001101001100100000 011000
0011011110001100100100011 111001
0001000110000111000010001 011100
0111000100101000001011100 000010
0010011001110010001101111 111000
1001101000111110100100110 11011110
1101011101000101010011001 011101
1110010111010001100100000 001101
0101001111001011110110111 0010000
1000001010100010011110011 11100
0000111011010001001010000 111001
0111101010110011001100101 10100
1101100010110000010010111 011111
n = 31

0101001101011001010101111 0001010
1000010001010001011100101 1111101
1010111111011111101011110 1010100
0011010011111110010110101 0000101
1011110001001110110101100 1100110
0110111101001010001000011 1100100
0110111010011110001100000 1110100
1110000111001111110100011 1100111
0111100001001110011001010 0111011
1010110101001110000110001 1010000
0001101100001000100001001 1001100
1000010010111100110101010 0001110
1001101101111110111111100 0000001

0100101111011101001101001011011
1101010111101001010011100101011
0111111000100001010011000000110
1000001001011100101000111000100
0100010011110100011110110110100
1110101100100011000011111001100
1010011110101011100010111011100
n = 32

1000101101101111100100111011111100
1111110100100001100101101110010 0
0111110101001110011100110001000 01
0011010000000110011000010011100 10
0000110011001011001000010111001 11
0010000011000101101110000000110 01
1110100011100010001001000011010 11
1100100101100111101100000101110 0
0111101111000100101000111011110 1
1000100001110010111011001001011 11
1011011010011011100011001000010 10
0101101101110010000110100011010 10
1101011101001000000010100101011 00
0000100001001011111100010110110 00
1001110101001100111000001101001 11
0000110101111010110100101000111 01
1010110010000110010101001011000 11
0011111001011101001011100010111 0
1111011011001001010111100101010 1
0110001100010101001101111110010 11
n = 33

0110011110101100001010111010010 10
0011111111101000110011011010100 01
0111101110011100001111101111110 10
0111101111001101011001011010110 11
1100100000111111001001000101100 01
0000000011010110010101000111110 001
0000000010010011111111111110001 000
0000111011100110010100110010000 001
1011011000101111111110110111100 01
0110100111111111001111111010010 01
0011100110100010011000000111000 010
1100110011101101101000001000110 0
0100011011100110011101000000100 00
0100101101111101010100010000000 101
1111110011101101000011101100000 00
0011100110110001110111010110101 011
1010100000010100111000110010100 001
0010011101010100000111101010110 111
0001011100111001110101111001101 111
0001011000101111010100111100001 0100
n = 34

1110100011101101010011000011110 0101
1100011111110111101010110111110 1111
1011100001011101000101001100100 0001
1100111110001101011000001101101 1000
1110001111011010010110001111101 101
1011111100000101111110101111010 101
0100000010010100011001110010011 100
1110110111010100000001110100100 0110
1000110011101001111011101010001 011
0011000000111000101101100100010 0110
1000001010110000101000111011101 1101
0001011111111101101110110110101 100
0111100111110100001111011011001 101
0000010010010011001010110101010 001
0001101011000101000000001100001 101
1000001010011110110000010000110 0100
0000000010001000111010000000100 011
0101010010001111101001010000101 0001
1011001000011011001000111000111 01
1101000110111001100110011011011 011
n = 35

1000000100110010001010100111010 0000
0011010000111110010100011010111 00001
1100101000101011011110001110110 10101
0111100100010000000000000110110 11000
1001001110101000000111011100110 00011
1100110010011010000010110101000 00011
1001111101011100111001001101011 01011
1110001000000001101010110101000 01010
0110101100101101000100010001111 10011
0100100110110110001010111001110 10100
0010011000011001000001100111110 11011
0001001001001100101111000001110 10111
0110111000100011001010100011100 11011
1101111010111011001000001110001 10110
0111000101010110010011100000111 01100
0111000010010100011111100111101 10100
0110101001101001010110100010011 01000
0011001010010110111010000101001 01111
0111111001100010100111001100011 01100
0000101110101010101000010101101 11110
n = 36

0111000010011110010110101010001 010110
1001010001001000110110111100000 111110
0010101001111110111011010110110 010100
0010101010110011111100100100100 0111
0010010101110100110001100100101 00100
0010110010000011111110101001001 111010
1011001100111000011000010001010 110010
0110011111001001000000110000000 000000
1101001010001111110001111101010 001110
1100110100000100110101000111000 01011
1110011010100011101001110010111 01110
0101110000010110011011100011010 01001
1100011101011100000111101000101 00001
1100110010110000010001111100001 111100
0110000101001110100011001010101 000011
1111111000110001100111011000010 100100
1000010111000011101001100010000 00011
0100110011010001011000000001000 11000
0011011010010101111001010011 001111000
1101000000010000000101110111011 000100
n = 37

1011111011001001110000100100011 001010
0111001001001101101000010111001 101010
1110001001101011011011100111100 11101
0100101101101110000010010111100 10000110
0100010111110110110110001001010 011010
1010010110111110110010101101110 010100
1010000110100101111011100010001 0010110
0111000010101110000000101011000 0000001

000011011101011111000010011010011110011
110110010101100110010011011110110110111
000100000101011100010101111111100100
010001101001100011010011101001100010011
101011100011101000010111010001011011
100100010111101111001111000100011001
000100001000011101111010000100110000101
000101011101010000110011000100100000100
111100000101101111010010101011001101
100010100101111010111001010101100001111
000111011100100101111001010010100100001
101111100100101011000100000101011010
n = 38

010111101010000111011000100011011000101
100110100011000000001100011100111001111
110111001010001110111001100011011100001
000100010000100000010100101000011110010
010111001100000101100100111101011011110
110000001010111011000000111100111100001
100110111001110111010000001110101101101
011101101011011010011110111000011101101
111101101110011010010101011011011110
011010111000110101010110000111001101111
101001110111110100011001111100001011111
000000101000111100111111101000110100111
011001011010000111111001011010010100100
000110000010111100011001010111010110011
010100011011001000010101100000101010000
101100011101000011100010011000111010011
011010011101101111011011111110010010001
010101110111110110111111110111010010110
000001011011001001111011001001111000010
110011001111110101111110101011100101010
n = 39

101111110111011111000001011010101001110
011011000110010100011011010011011000110
001000100111010100111110100010011011001
100101000011000010100010111110001001101 0
100110010110111101110111011010000011
001001111001100000101101001101001100100 1
111111011011001111010010000000010010
111101011011010111110011100111100011001
000100000010101011100100101011011110000
110010110010000010001110010110100000100
011111110011010101100001100110011111111
110011001101111010110001011010100100101
001101010101110100001100011010000101110 0
110101001011101100011100100111010100000 0
000101111011011111011001101010100001110
010001100111000010101100111010001111111 1
111001001010110010010011101000111111111 1
010010011001111011111011010111010110000
001101100111110101101010101011111101111 1
100101101000010011011010010011000010111 0
100000100000100111010110000110101000000 0
010101110111111011001101001100010100011 0
111100100101011001001001110100011111111 1
010010011001111011111011010111010110000
001101100111110101101010101011111101111 1
100101101000010011011010010011000010111 0
100000100000100111010110000110101000000 0
010101110111111011001101001100010100011 0
111100100101011001001001110100011111111 1
010010011001111011111011010111010110000
001101100111110101101010101011111101111 1
100101101000010011011010010011000010111 0
100000100000100111010110000110101000000 0
010101110111111011001101001100010100011 0
111100100101011001001001110100011111111 1
010010011001111011111011010111010110000
n = 41

101111110000010010001110001011011010111100
110000110100000010011101011010011110001 10
000011010011101000011100110011111101010 00
111111110100111010111101100110110111010 00
001101110011110000010100001011001100110 10
001111110001100101010101101000011001001 010
111010100100110111010100100110001011010 110
100101010010011011010100100110000010110 10
100001000001001101011000011010100000000 10
010101101111111011001101001100010100011 0
111100100101011001001001110100011111111 1
010010011001111011111011010111010110000
001101100111110101101010101011111101111 1
100101101000010011011010010011000010111 0
100000100000100111010110000110101000000 0
010101110111111011001101001100010100011 0
111100100101011001001001110100011111111 1
010010011001111011111011010111010110000
n = 42

100010010110111000000111000001010111010 1011
100000101001000111101000110001010110100 101
000100100011001001110000010100111011010 0100
101101100010111000001001111011110001100 00000
000101110000111000001111010010100110001 10111
010011110100001110011111110101000001011 1111
000100100100110001001001111000110000100 01101
000101100101100111001011010101010110011
011010110010000100010001010100111111110
110000101011000000011100011000101110010 010
001010000100001111000101100111100011000 001
011100011010100010010110000001101111000 0011
000111101101100100011010101110011001000 001
101110111110000111111110111010011111010 01
000110001101011011111001001000010111101 11
110110110001101010001001000101100100000 010
011011110100101000111000111100111010101 0000
110100101011000111000111000101101011010 111
111000001010111010101110111011011010000 0
000101110101010010000100100001110101110 010
100000010001100011001001111101111101111 011
010000110111001010111001110000011100000 001
100110000110010000001000111111110101011 01
111011111100011011111101000010010000111 10
011010101111110010001100000010000101001 111
000110000110110001111110101001011110001 110
100010011001001101000001001101110000110 1101
001000001001011011110011011100000111011 00
001100111001100100010101001001000110 10100
110000110100011000110100101010110010010 110
n = 43

110110111111000101101010000001101011001 0010 10
000000101101001100011000111000010110010 0000
111011111000010000110011101011000001000

010101101001100000110000000000001111010110011
001100010101100011100011101010110111011100101
100010011111001100110100110111000100010000110
000111111100110101010100110000010000100000110
001111101100110110001001000111010000001001010
000000101011100010000100100000111000110111111
011100101010110001001101110011011000100101111
011111001010110010110111000010101010010010101
001110100101010110001011000000000100110011100
001110100110110001111001010000101110010111101
101010100101000111000011110110001011110100000
111010001111110011111101001111011100000100101
111001101000000111101101100111000111111100101
000010011001011101000100100001111001011100101
001000100010111011110001101000111010110010000
010100101010001111100010010010011010100111111
011000000101010111010100010101110101100111111
n = 44

001101111110110110010010111101000110000111110
111110110000001101110100011011000001001001010
001100010100110000111011010011001101101111000
011101111000101001001011001100100110010110110
000111010010010011011010100111001111100011001
101100010010000011010010101010001010101100111
101010001001101010010010011001010000110001011
001110000101000101010111111100000011010100111
111101110010101101000110111101111011000011100
111100000010100001101010010000100001001111000
010100100111101111111010110001001101100010101
010011010010111111011111110110000111001011000
011110001101010110000001010010111001100011010
001001010110001001001110110000111111100100011
000110111101000100101001101101110001100010001
101010111000000101010100011101000100110111111
111011000110101011001100110001001111100111101
101000000100010111001011001000011001010010101
011010000111011110111000000100011111100111000
101000101001110111011000110011001000101101011
n = 45

0100010000110010000010100000011110100001001100
1101100000000111110110011111100100110101101101
0010010100100111010111001101101000100100001010
1000001110100111110001001100110011000001111010
1000101010001111011101111011001100001011010101
0101110110000101111110011000001010110101010100
1111010000101111000000000111001100011000111010
1101001010111101100110010010010000111101010110
1101111111111010001101001100000111000001010010
0000000011111011111001001100010110100011011100
1101100110110111011111001000100000000100110000
1010100111110111101110100100100100110010000101
0011001101011111001100001000100111101011101110
0110111011111100101010000001101100101010000011
0100110000000111011110000101111000110000001110
1111110001101011101110000110101101110111111101
1110110110001111011111011010000100110010010111
1110010010011011111001000100010111000110110110
0011001100110010010111001010011000010000101001
1010101001100011011100100101010011010010111100
n = 46

10110011100001010111001011011100111101010000111
00111101010111011001101010111001000010010010001
10010010001000111100000110010001101101100100010
10010000000100111000110111100001010001011101101
11000010110011101110001110011100001010111000110
11111100011001110110010001001100000010100100111
11111110100110001010111111001001000110010010011
01011000101001001101000100110011100110100100000
10110110110101001000011111011100010111111000101
10010101000101101110010011111101110001100011110
01001100101100101011001010011100010011010100110
00001010110000111110100101011010100000010101111
10100101110111001110011100111111100101110000011
01100100100111010110110001110011100101011011111
00011111100001111100001001000000110100000111110
00000101100010101111100010011110111001011111111
11110011101111110000000001101100100110000001010
01100011001110110101111100111010110000101000011
11100101100101000000100000110111001001010111110
10010101100101110111111000110101110001111100001
n = 47

101100011111110100100000110010101101010001010010
100111000001101110010001010000100010111110000110
101000010100000101110110100001111010010111111001
010100001100101000011111100100101011100101001110
000101101011111000110110111100010011111101110001
111110010010111101111110111011001011110110011000
001001100100011000101000110000101001000000000010
100011111101111110111111100000001100110111001000
000101110001011010101010000110101111011000100011
010101111100111110011100011110100000011111100001
000101001010001110100101110001100010000111101010
001000101110101111011100010000110100001001011000
100011110100100101011110110100110011100101100000
001001011010010000100011011011011111001011000100
101000111010101011111010001000010111110001010111
001111101100110100001000101111100100101100100111
000001100000111010011100101001010000000001001011
110001000101101111001010010111001101000110101111
111110110101101000111010111111001111001000101011
011110011011010111101110101111011010110101010010
n = 48

0110110001110101000000101111110000010111011101011
0111101010110001100000011111000000011010011110001
1001111011010011010011011001011100111111110101000
1100110010000001111001001111100101110101000000010
0000000011101111101010000101111000001100000101010
0000111111000001100110010110110011011111010011101
0101010100000111010110111110001110001101100111000
0001010010001110001000000000101001100010100100111 0
1111101010000100001101001000101001101011011100111
0100111011001001011000110111001001100111011110111
1001111001000100011101001000000011110000111111000
0111010100101110101010110100000111101000011000100
0001001010000010100110111111100111001001111101101
1010101110001000111010001011110111000001011000000
0100111010111011001100100101100001100011110001101
1100110100110110111100101111011101100001010010101
0110111101111110111110110100001011001010110101100
0101100110001000100111001011000101101011100011111
1000010110010101101011111111000000000101101100101
1100101000011010011110010101101001010000010101001

n = 49

11010010010011001111010100000011100011010000100111
01011000011110001101001000110111000000001101100100
11001001000100100111101010000111010101101111111010
00001100100100011100100110010001101111110110011101
10010001100110111110000010000111001101101110101101
01000000000001101111110010000011001001100110001101
00100101111000010111010100111101101101100011111011
01100101010011000001011000001101000010110001000100
00011101010000111110001100111110011101011010111110
11011001001001110011100100101100000011001111011110
11011100001110101110100101001001101010010101000110
11110010110000011101111000010010001010011001110001
01001000010110110100101011010110100010000101100110
00000001010001010100011110110010000111111010111000
10000100111100001101001110111001100001011000011000
10101011001001001000011000000101011001110100010010
10111001001000100110101000101110111001010100001110
01011111010011101011111100011001010100010000001011
01101101110001000110110001011110011100010110101100
10000010011100010010011100111001100110010001000000
n = 50

001101011011101000010110000011011011001101100010011
010001010010001010010011000111100101010000001110001
111001101000000011111010100010000110100101001001011
000100100100001110111100011010010010001111101011111
001001001011110101101111100000100010101110100101100
010101101101111011101011111000111010101100110011100
101010100101100111100001011100111101011110110101000
000010001011100011101010000010000100101101101101100
100011110011000101111011100010111110111011011011010
111001001010110110011111000010010001101010011100011
001001000001110010010100110111101111011001010100100
000100010100011101001110100010111001010100000110100
100111000100110011110010101101010001010111010000001
110100000110110001100001111010000100101001101100000
110000110001010011001011100011110001011101111110110
011101101000000011110000011110011100111100010000000
100000010000000000100101100111010011111011010001001
001110000101000110100111001111011011001000010001010
010010101101100011111010110111110010000011001111100
011111101001001100111110111101100110110111110001011
n = 51

0010000000011010100101111111110101100101101101100100
0010001010000000010111110101111110010111000101001100
1010010010111011110011010101111100110011110101100111
1011110010111010001000100011011110110001101010010001
10000001101000010000101100110010110000000100001111001
1100010110100010100101111111100100111001001011101110
10110111011010111001011010000111010000000001011011001
1000101110000101111000100101000011100011010111100100
1011100011101010110100110001110111000101011000010000
10111111000100100011001011011100111010101110000011111
0001100111001111110011010011100100011010010110101101
1010101101111100000100100011010101110101111111011101
0111101001011110011000111001010110011111101010011011
01111000101100101000110111111011001111100110000011110
0100011101100101000010100001110010111001000011000101
11011000111111100010011000000001010001011011101000010
01101101000011010011101111110000111100011011110011111
11110000001001001110110110000011100000010100000110011
1110010000000011011101001100011001011110100100011000
1100011000011111000100110110110001101101000101100000
n = 52

00101011000001100101110010101001100001010101010001101
00011111110101001010100010111101010000101000100000011
01110011100110111000100101000110000101000110000110101
10100010100110010011011011101001001001001100011001001
00101110011001111101000010010100100000111001000100101
11010110011100000000100001110011001111010001110010010
001000101100010100010111111111000010101001001100101110
11001100111000100111100001000101001101011110111000010
11001110001011001110101000111100101100011100000011101
01111010000110101110100011001110111110011000111000110
10111010110011001010100010111000110001101010011101100
10101001111010010110110010111101110001111110110101010
11110100011010000010110111101010111100010000011001110
00101110110110111010110110101110000101101111001110010
11010010011010111000010000101111111010110100111111100
10010110001010000111001000100001000010010011100101000
10011101001111101010010111001110100011011011101000110
10000011100111001010100110010110001101010100000110110
11010100010101111010111011101010010011101110001100000
00010101000011111101111010101100100101110110001101011
n = 53

101001011000000001010011010000101001011011110110101001
000101010011101001001011111111001110110111110001010001
010110110010000011101001001001011111011101001110010010
101101000100100010011010111110000001010110001001100001
000101110011110100111110101010011000011010000101101011
011110100100001011011001101111111100011001110100001100
000000011110011110011000001111010101111100010100011111
1001111100101111011110101110011000011111110110011010011
110101110000111001000011110100010001010101000011110111
1110111111100111010000011111010010101111011110001000110
110001000111011000111010000100011001100011001100011011
111101010000110111001111100000110001010101001101111000
111101001110111010111000010110000110011101011110010111
110011011011010001001011011100000011001100010110001110
101000110111100001010100001101110100100111101101001000
110111011110111110000101111010111001001011011000111111
0111010001011000110001000000010001001011101100000000100
101111001011110000011001110000111001110110100110001000
110011011000110110101101101100001100010111101100101011
010011100111011100001001010111001001101011110011011110
n = 54

0101000111011101000010100010100110111111011000001010101
11110100000001011001110010011011001011010001000100110001
0010100011101001001110011110011100011011111100011111000
0001011110000101110011101100001111010010111100010101011
011010000000010110001011111100011100110010001100001000011
01100101001000000101111010011000000110110101101100101000
1011100001011000110000101010101001001111001101010001111
0100010001101101001101111000011010010101000000111111111
10111000010111100100100000000001110001001000101100010110
1110011110110011011011111101010111101001101111110100100
0110011100001110011100111100010011001111111010001000010
1011001001101100111111110100111101011010110111110000100
0010111100101100000100011111111100101010101110010001110
11100000001011010001110110111000100101001110010101100010
1111011110110000100100011010111011101000110011000000111

0000101100010110110011100001101111101000110010100110101
0000111101010000011010000111110111010111111011001111011
1101000101111000010111110001110101111000010111010010011
0100110011110100110011001111001101000011010111010111110
0010011101110100111000010101110011001100011100011000101
n = 55

10100101110011101101100110011100101000011111011110000011
011110101100101011111111001101110010100100100011100000000
0000111010111000100110001110010111010111110111011111010
00101000011111001111110011011100001100110010011100111111
00011100011110101011100111000011000101100100111000011100
000100000001000000001111011000000110101010100110011000100
011000101010011100001110110010000000010000001111101111000
01100100000110000111001110010111110110101010010101011010
01100001010111010001010101100000010001011111101001001111
00101100011111101001011001001001111101100100110111011011
100000011000000111001111010010010011011111100001010010000
01011011000001110100111111111101101011101110100011100101
11110001001101101011010011011100101111000111011101011010
10111011101101000011100000000001101010010111100011110011
01000000111110011010100110101101000110001001111110001111
10011001101000100101110100110100100011101101011100010101
001100111101001000100011001101101101001000000000101011110
00100110011011000110100010000111110001011111001011010100
11010111010010011001011001000110111101100111001101111101
100111011110100111101100100011110000000001000101110110001
n = 56

110001011010101101100010110011001000000000100000011101100
010111101110010110001010110111001001010010111000010100000
100010111100011001000011111100010010100011100111100101010
001001111001000000111101011110101100110011100110011110111
011111111111111011101010011000110100101101011000010010100
110001001000010010001101110100110111000001110110000011101
0110110011010101000100000000100100011001101101000110111111
0101110000010010101111010000000101011001001010011011000111
111000100111011011101001111001100100000010100110101110001
011100101110011111100011100010011000111011011010010001110
0000011011111010001001100100001111100111111010110110011000
010011101011010100001011000011011100011101100111011000001
110001000110011101011101110100110011010001010100000010010
0110011011111110110010000001101100001011100000011100101101
0101110001001011000101101011001010111100110111111011100001
011110101000100101110011000110100101011000111111010100011
010100001001110111000110010110101111110001110111100110100
100101110010110001101101000101100100111101110110001101110
111000111101001001011111110110011010111101111000110101001
000111010101000110000001111101010111101100010100001000101
n = 57

0010001001110000110000000011100100010111001010001111100001
0001010000110110101000011001000101001000100111101101100111
0011111101110100010011010000011101100101111001011010000011
0100010111101101011010110111010000111011011000111100010110
1100100100001110010001101100110000010111110010011100001111
0101010110000000101110111111100100100111100110101110110011
00100011011100011110011010110101011100110111110010010010011
0010010011010010001100101111111011010100100010001101111010
1111000111010101000010001110010010110010010011100111001010
0011101111101010010111000011111000110000110111011111100111
1110000101000101101011011000100110111100011101011001011000
0110010100111011101110111111100111111001100111111011010001
0001100000110110011101011100011001101001001100000011101101
0111110111110011011010101101001100101110001000001001010001
1101111111100101111100000000101000111010101111010101000010
0011111111110011110011111011101111001000000100001110101100
01111101110011110110101010101111000101011010001101011011011
00111001110111000000001000101101110111000011000110111010000
1001000011111000111010101001111000001010111100111101101010
0000101111101111110000100111011110001010111100100101100100
n= 58

0011000101011100001011111111111111011111010010100001001010
11101011100100010010100111011010110100011011111100110110110
011101010111110000010000000010011000100001100111100110110011
00110000110001011010101111100101010000110000011111000100110
1011001001110111001101111111101001110111000010100110000001
00110000111101000010000010001100111110001001011101001100100
11000000000111011001110100000100101010010011101110100001101
01100111001100011111011011010011010101101110010001111100000
11100010110001100100101101111111101001010110001111011100110
01110000111000101011001011100011110110100001000100000100101
10000111000111101111000100011000101000001000110011011000001
01100100101101010101111110000010101001100001111010000110110
01100101010101111000010100100010001111010100010111110001111
00101011100100011111001100100000000011001110111000110011101
00001100101010001101011110110100110011100011000111011011110
10100101011110001000101110010011001100101101100001100100011
011010100110100001000111001000001111011111101000000011110000
101001111111100001101001011001000000110100110011000111001000
10001011111001110010111001100100101001111000000110001100111
00001110111011101001010011111001101110010000101001100000011
n= 59

010101110001011001010100111111110010010011111000011000110111
000101110001101100001100101000011110000100101100100100010101
101010011010000110000001101101001111111110110101001111010100
100110111101001001110100011111000010011010001100000101011111
010001111011000111010100101000001110011111100110111010001001
010000010001111101000000110000111101100010101110101010001000
011100011101111011101000010111010010001110001010010101100001
01111011100110010000111110111110111010111000011111110010001001
110010010000001100000011100100001101101000000000111000110011
100100101010011000001010010111101110010100100010000111001001
000100000011010111100011110000100101010111010001111000101001
100001111000010010011101000100001111001010101110000100110000
001001111011010011100000010000100001001011001110011100010111
111101100010000010000111100110011010001101010011001101000100
011101001001001000111010110010101001110101010100011000111011
011011110010010000000010110101100010010000000100011010010011
010010111111111011000100100010111000000011010010000010101010
001100111100011111100000110000011110011001001001111111100011
0010010001101111110110111101100000000001011011001011000001010
011100000100110010011000110111010010000001010001100011111011
n= 60

0100101100010110010011100101011110011110100111011101001100111
00100111010010010101000101100011000100000001010110000010100010
011000011001111100001111000100111101011001111011100010111011
0001111101100001010011110000100101000010001111111111100001111
0100001010011000000010011110100100001001110110001100001000101
0110010011000010000010001010000101010100100110101110011000000
0110111100101010100000111010000010101001000100010001110101101
00010101110111111111110010000111100111100010110010001011011101
0100001111010110101011110110000110110010000110001111110001110
1111001101111111100001010110011001101100001010100101011000010

1111011010011100011001100110001001001011011001001011010111010
1110111111110001100111001010110011001010101111011100101100111
1000101000111110100111000001111111111010111010101010110000001
0111000111100010101000101010101001100011000111010000011001
0111111000000110011100110101110110011001100100010000011011100
0000101010000011000111001000011011011001101011010011001001000
0000100100111100000011001101100101110100001101000110011110011
1010100011010001101010110111001001100000100111010101010001000
0010011101111111111011001100101010111010010011011111001101
0111110100011011001001100111001010000000111001010010111001000100
n= 61

1110001111011000011001000001000001011010001111000010101000011
0101111011001100100011000111000011000100111101111010110000
0010100010110001100110000010100100001100001001010011011011011
0010110000010111001111000100011001001011010010011010111001010100
0000101111000011110110110000110101101110111001010111010001001
1000000111111001000001000011001101010111111001001101111001000
1011101000010001011110010101001101010001110111011010111011101
1101001100011101000000011101110000000111001111001000001110000
1111011110101010001110010101000111011100011111010000101110010
1011010010111011101010100011011001110001110101011010011010110
1100011001100010110010010001111100000101100000101110101100111010
0110011101010000101011010100001010011000111000000111010101010
1101001100110010001011011100001111111001001100010001011100000
1011011110101111010111111010110100000001100000000010111001
1101000100101011111001010111000001100101100100101100001000011
0100010000101101011001100101011010001111010110110110110011010
0000110000111010010010101010000011011110100111011011010011111
0001110111011111011011001011100001100100001110010011001100110
1101010010100011111101100100100101100011100000001111100011001
0011111100000010101011100101011111010011011101110001111100011
n= 62

0111000001110111010001101110011101011100100101100110110001100
1000100110011110101110000110111010110111010111010011011010000000
1110000100110001130010100010101010011001110110101111011110001100
1000110000000011001010010110010010101010110001110110110011001
0010110000110111000100111010101101110011000010100000000011111
0010100111101101100000011111110110011010000001001100110011001
1010111111010110010000010111100110111001011000100111000101010
1100011110101000010011010001101110111010110100111001100101000
1101010111010001001001001100001110100110100011100000001000011
1010110110101110011111001000011111010110100100100011111010010
0011011000010111100110011011100110001100011001000011110110001111
1000001011010101001100100111001010001100010111110100100000001000
0010011101001000011001100100100110111011000001011010011100000
0000001000000011011000001001011111111000100110111111010000101111
0111111010000101100100000100001011100000110001010110101101010
0001111110100000110011000100011110101101011100010110101000000000
1111001110010100000001100101100100101001111001100010110111001001
1001010111110110100100100101010111110101000011101111100100101
0101010010011101000100110101000111001111111010000010100110000
0110111000111101011100011111100001100100101110001001011000110110
n= 63

1011000000000111100101100010111000101001000011101110011000010
0010100000010001111001010101010000001110100011001010011011000001
1110110100011000001111111111001101010111110001011011001111101111
1101001101110100000011011100100100010010111010110101101011010111
1001100000000011101101100100010000011011111001110000111011101
1110000010101100001111011010000011011100101000011101000000101101
0101100001000110111111110011011000001010100111011010110110101101
1111100101101101110010110100001011010111000101110101001001111101
0100011101111100101111001111000011011001101010101011100011000
0111001111011010011101100010111101011111010101011111000011101
0100001100000000111010100011010010000001011101110001001110101111110
0000111111111011001011010100101100011000011001010110000011001011110
0110100001101100101001010111000011001011011101111010000011010000101
0010110100100011110110011001100101111001111010000001101000010011
0110001110110100001000100111111111010110110101100100011100100110
0010000010110111111011000011001000100010100100011010001 1111001
1010111110000101010111000010000011100000111010010101011001001001001
0011101000111110101010001000101111111000111000000110000100110101
1010111011110111001101000001110110010011000100010001111010011010
1100100000011100000001010101011100010100000111110111111110110011
n= 64

Código - C

Brute Forces

```c
void showbits(integer_t x, int total_bits)
{
    for (int i = total_bits - 1; i >= 0; i--)
    {
        putchar(x & (1u << i) ? '1' : '0');
    }
}

void showbits2(integer_t x, int total_bits)
{
    for (int i = 0; i <= total_bits - 1; i++)
    {
        putchar(x & (1u << i) ? '1' : '0');
    }
}

int bruteforce_rec1(int n, integer_t *p, int currIdx, integer_t partialSum, integer_t desiredSum, integer_t mask, integer_t *comb)
{
    if (partialSum == desiredSum)
    {
        *comb = mask;
        return 1;
    }

    if (currIdx == n)
        return 0;

    if (bruteforce_rec1(n, p, currIdx + 1, partialSum + p[currIdx], desiredSum, (mask | (1 << currIdx)), comb))
        return 1;
    else
        return bruteforce_rec1(n, p, currIdx + 1, partialSum, desiredSum, mask, comb);
}
```

```c
int bruteforce_rec2(int n, integer_t *p, int currIdx, integer_t partialSum, integer_t desiredSum, integer_t mask, integer_t *comb)
{
    if (partialSum > desiredSum)
        return 0;

    if (partialSum == desiredSum)
    {
        *comb = mask;
        return 1;
    }

    if (currIdx < 0)
        return 0;

    if (bruteforce_rec2(n, p, currIdx - 1, partialSum + p[currIdx], desiredSum, (mask | (1 << (currIdx))), comb))
        return 1;

    else
        return bruteforce_rec2(n, p, currIdx - 1, partialSum, desiredSum, mask, comb);
}
```

```c
int main(void)
{

    int func = 2;
    // 1 - > Recursiva
    // 2 - > Recursiva OTIMIZADA

    if (func == 1)
    {
        for (int i = 0; i < n_problems; i++)
        {
            if (i > 27)
                continue;

            unsigned int n = all_subset_sum_problems[i].n;
            integer_t *p = all_subset_sum_problems[i].p;
            double somaTempos = 0;
            double tempoMaximo = 0;

            for (int j = 0; j < n_sums; j++)
            {
                integer_t desiredSum = all_subset_sum_problems[i].sums[j];
                integer_t comb = 0;

                double time = cpu_time();

                if (bruteforce_rec1(n, p, 0, 0, desiredSum, 0, &comb))
                {
                    showbits(comb, n);
                    printf("\n");
                }
                else
                {
                    printf("\nSoluçao NÃO encontrada!\n");
                }
                time = cpu_time() - time;
                somaTempos += time;
                if (time > tempoMaximo)
                    tempoMaximo = time;
            }

            double tempoMedio = somaTempos / n_sums;
            printf("%d %f %f\n", i + 10, tempoMedio, tempoMaximo);
        }
    }
```

```c
    if (func == 2)
    {
        for (int i = 0; i < n_problems; i++)
        {
            if (i > 27)
                continue;

            unsigned int n = all_subset_sum_problems[i].n;
            integer_t *p = all_subset_sum_problems[i].p;
            double somaTempos = 0;
            double tempoMaximo = 0;

            for (int j = 0; j < n_sums; j++)
            {
                integer_t desiredSum = all_subset_sum_problems[i].sums[j];
                integer_t comb = 0;

                double time = cpu_time();

                if (bruteforce_rec2(n, p, n - 1, 0, desiredSum, 0, &comb))
                {
                    showbits2(comb, n);
                    printf("\n");
                }
                else
                {
                    printf("\nSoluçao NÃO encontrada!\n");
                }
                time = cpu_time() - time;
                somaTempos += time;
                if (time > tempoMaximo)
                    tempoMaximo = time;
            }

            double tempoMedio = somaTempos / n_sums;
            printf("%d %f %f\n", i + 10, tempoMedio, tempoMaximo);
        }
    }

    return 0;
}
```

Horowitz and Sahni

```c
void insertion_sort(integer_t *data, int first, int one_after_last, integer_t *bits)
{
    int i, j;
    for (i = first + 1; i < one_after_last; i++)
    {
        integer_t temp = data[i];
        integer_t temp2 = bits[i];
        for (j = i; j > first && temp < data[j - 1]; j--)
        {
            data[j] = data[j - 1];
            bits[j] = bits[j - 1];
        }
        data[j] = temp;
        bits[j] = temp2;
    }
}
```

```c
void Merge(integer_t *data, int first, int one_after_last, integer_t *bits)
{
    int i, j, k, middle;
    integer_t *buffer;
    integer_t *buffer2;
    if (one_after_last - first < 40)
        insertion_sort(data, first, one_after_last, bits);
    else
    {
        middle = (first + one_after_last) / 2;
        Merge(data, first, middle, bits);
        Merge(data, middle, one_after_last, bits);
        buffer = (integer_t *)malloc((size_t)(one_after_last - first) * sizeof(integer_t)) - first;
        buffer2 = (integer_t *)malloc((size_t)(one_after_last - first) * sizeof(integer_t)) - first;
        i = first;
        j = middle;
        k = first;
        while (k < one_after_last)
            if (j == one_after_last || (i < middle && data[i] <= data[j]))
            {
                buffer2[k] = bits[i];
                buffer[k++] = data[i++];
            }
            else
            {
                buffer2[k] = bits[j];
                buffer[k++] = data[j++];
            }
        for (i = first; i < one_after_last; i++)
        {
            data[i] = buffer[i];
            bits[i] = buffer2[i];
        }
        free(buffer + first);
        free(buffer2 + first);
    }
}
```

```c
void somas(int n, integer_t *vals, integer_t *sums, integer_t *bits)
{
    integer_t som, comb;
    for (integer_t a = 0; a < 1 << n; a++)
    {
        som = 0;
        comb = 0;
        for (int i = 0; i < n; i++)
        {
            if (a & (1 << i))
            {
                som += vals[i];
                comb = comb | (1 << i);
            }
        }
        sums[a] = som;
        bits[a] = comb;
    }
}
```

```c
int horowitz(integer_t desired_sum, int sizeA, int sizeB, integer_t sumsA[], integer_t sumsB[], integer_t bitsA[], integer_t bitsB[], int nA, int nB)
{
    sumsB = &sumsB[(int)sizeB - 1];
    bitsB = &bitsB[(int)sizeB - 1];
    while (sumsA < &sumsA[(int)sizeA] || sumsB > &sumsB[0])
    {
        if (desired_sum == (*sumsA + *sumsB))
            break;
        else if (desired_sum > (*sumsA + *sumsB))
        {
            sumsA++;
            bitsA++;
        }
        else
        {
            sumsB--;
            bitsB--;
        }
    }
    if (desired_sum == (*sumsA + *sumsB))
    {

        integer_t b = *bitsA | (*bitsB << nA);
        for (int i = 0; i < nA + nB; i++)
        {
            printf("%s", b & 1 ? "1" : "0");
            b = b >> 1;
        }
        printf("\n");
        return 1;
    }
    else
        return 0;
}
```

```c
int main(void)
{

    for (int i = 0; i < n_problems; i++)
    {
        if (i >= 0)
        {
            int n = all_subset_sum_problems[i].n;
            integer_t *p = all_subset_sum_problems[i].p;
            double somaTempos = 0;
            double tempoMaximo = 0;

            int nA = n / 2;
            int nB = n - nA;

            p = all_subset_sum_problems[i].p;

            int sizeA = 1 << nA;
            int sizeB = 1 << nB;

            integer_t *sumsA = malloc(sizeA * sizeof(integer_t));
            integer_t *sumsB = malloc(sizeB * sizeof(integer_t));

            integer_t *bitsA = malloc(sizeA * sizeof(integer_t));
            integer_t *bitsB = malloc(sizeB * sizeof(integer_t));
            double time2 = cpu_time();
            somas(nA, p, sumsA, bitsA);
            somas(nB, p + nA, sumsB, bitsB);

            Merge(sumsA, 0, sizeA - 1, bitsA);
            Merge(sumsB, 0, sizeB - 1, bitsB);
            time2 = cpu_time() - time2;

            for (int j = 0; j < n_sums; j++)
            {
                integer_t desired_sum = all_subset_sum_problems[i].sums[j];

                double time = cpu_time();
                if (!horowitz(desired_sum, sizeA, sizeB, sumsA, sumsB, bitsA, bitsB, nA, nB))
                {
                    printf("\nSoluçao NÃO encontrada!\n");
                    break;
                }
                time = cpu_time() - time;
                somaTempos += time;
                if (time > tempoMaximo)
                    tempoMaximo = time;
            }
            free(sumsA);
            free(sumsB);
            free(bitsA);
            free(bitsB);
            double tempoMedio = somaTempos / n_sums;
            printf("%d %f %f\n", i + 10, tempoMedio+time2, tempoMaximo+time2);
        }
    }
    return 0;
}
```

Schroeppel and Shamir

```
typedef struct
{
    integer_t som;
    integer_t comb;
    int ind1;
    int ind2;
} sum;

void insertion_sort(sum *data, int first, int one_after_last)
{
    int i, j;
    for (i = first + 1; i < one_after_last; i++)
    {
        sum temp = data[i];
        for (j = i; j > first && temp.som < data[j - 1].som; j--)
            data[j] = data[j - 1];
        data[j] = temp;
    }
}

void Merge(sum *data, int first, int one_after_last)
{
    int i, j, k, middle;
    sum *buffer;
    if (one_after_last - first < 40)
        insertion_sort(data, first, one_after_last);
    else
    {
        middle = (first + one_after_last) / 2;
        Merge(data, first, middle);
        Merge(data, middle, one_after_last);
        buffer = (sum *)malloc((size_t)(one_after_last - first) * sizeof(sum)) - first;
        i = first;
        j = middle;
        k = first;
        while (k < one_after_last)
            if (j == one_after_last || (i < middle && data[i].som <= data[j].som))
                buffer[k++] = data[i++];
            else
                buffer[k++] = data[j++];
        for (i = first; i < one_after_last; i++)
            data[i] = buffer[i];
        free(buffer + first);
    }
}
```

```
void insertMax(sum heap[], sum soma, int size)
{
    int i;
    for (i = size; i > 0 && heap[(i - 1) / 2].som < soma.som; i = (i - 1) / 2)
        heap[i] = heap[(i - 1) / 2];
    heap[i] = soma;
}

void insertMin(sum heap[], sum soma, int size)
{
    int i;
    for (i = size; i > 0 && heap[(i - 1) / 2].som > soma.som; i = (i - 1) / 2)
        heap[i] = heap[(i - 1) / 2];
    heap[i] = soma;
}
```

```
sum removeMax(sum heap[], int size)
{
    sum r = heap[0];
    int i2=0;
    int i = 0;
    while (i * 2 + 1 <= size)
    {
        i2 = i * 2 + 1;
        if (i2 < size && heap[i2].som < heap[i2 + 1].som)
            i2++;
        if (heap[i2].som > heap[size].som)
            heap[i] = heap[i2];
        else
            break;
        i = i2;
    }
    heap[i] = heap[size];
    return r;
}

sum removeMin(sum heap[], int size)
{
    sum r = heap[0];
    int i2 = 0;
    int i = 0;
    while (i * 2 + 1 <= size)
    {
        i2 = i * 2 + 1;
        if (i2 < size && heap[i2].som > heap[i2 + 1].som)
            i2++;
        if (heap[i2].som < heap[size].som)
            heap[i] = heap[i2];
        else
            break;
        i = i2;
    }
    heap[i] = heap[size];
    return r;
}
```

```
int somas(int n, integer_t *vals, sum *sums)
{
    int i;
    for (integer_t a = 0; a < 1 << n; a++)
    {
        for (i = 0; i < n; i++)
        {
            if (a & (1 << i))
            {
                sums[a].som += vals[i];
                sums[a].comb = sums[a].comb | (1 << i);
            }
        }
    }
    return 1;
}

int schroeppel(integer_t desired, sum minHeap[], sum maxHeap[], int nA, int nB, int nC, int nD, sum sumsA[], sum sumsB[], sum sumsC[], sum sumsD[], int nP, int nS)
{
    while (nP > 0 && nS > 0)
    {
        if (maxHeap[0].som + minHeap[0].som == desired)
        {
            integer_t b = minHeap[0].comb | (maxHeap[0].comb << (nB + nA));
            for (int i = 0; i < nA + nB + nC + nD; i++)
            {
                printf("%s", b & 1 ? "1" : "0");
                b = b >> 1;
            }
            printf("\n");
            return 1;
        }
        else if (maxHeap[0].som + minHeap[0].som > desired)
        {
            sum r = removeMax(maxHeap, --nS);
            r.ind2--;
            if (r.ind2 >= 0)
            {
                sum s = {
                    .som = sumsC[r.ind1].som + sumsD[r.ind2].som,
                    .comb = sumsC[r.ind1].comb | (sumsD[r.ind2].comb << nC),
                    .ind1 = r.ind1,
                    .ind2 = r.ind2};
                insertMax(maxHeap, s, nS++);
            }
        }
        else
```

```
        }
        else
        {
            sum r = removeMin(minHeap, --nP);
            r.ind1++;
            if (r.ind1 < (1 << nA))
            {
                sum s = {
                    .som = sumsA[r.ind1].som + sumsB[r.ind2].som,
                    .comb = sumsA[r.ind1].comb | (sumsB[r.ind2].comb << nA),
                    .ind1 = r.ind1,
                    .ind2 = r.ind2};
                insertMin(minHeap, s, nP++);
            }
        }
    }

    return 0;
}
```

```
int main(void)
{

    for (int i = 0; i < n_problems; i++)
    {
        if (i >= 0)
        {
            int n = all_subset_sum_problems[i].n;
            integer_t *p = all_subset_sum_problems[i].p;
            double somaTempos = 0;
            double tempoMaximo = 0;

            int nB = n / 2;
            int nD = n - nB;
            int nA = nB / 2;
            nB -= nA;
            int nC = nD / 2;
            nD -= nC;

            sum *sumsA = malloc((1 << nA) * sizeof(sum));
            sum *sumsB = malloc((1 << nB) * sizeof(sum));
            sum *sumsC = malloc((1 << nC) * sizeof(sum));
            sum *sumsD = malloc((1 << nD) * sizeof(sum));

            somas(nA, p, sumsA);
            somas(nB, p + nA, sumsB);
            somas(nC, p + nA + nB, sumsC);
            somas(nD, p + nA + nB + nC, sumsD);

            Merge(sumsA, 0, (1 << nA) - 1);
            Merge(sumsB, 0, (1 << nB) - 1);
            Merge(sumsC, 0, (1 << nC) - 1);
            Merge(sumsD, 0, (1 << nD) - 1);

            for (int j = 0; j < n_sums; j++)
            {
                integer_t desired = all_subset_sum_problems[i].sums[j];
                double time = cpu_time();
                sum minHeap[1 << nB];
                int nP = 0;
                for (int a = 0; a < 1 << nB; a++)
                {
                    sum s = {
                        .som = sumsA[0].som + sumsB[a].som,
                        .comb = sumsA[0].comb | (sumsB[a].comb << nA),
                        .ind1 = 0,
                        .ind2 = a};
                    insertMin(minHeap, s, nP++);
                }
```

```
            sum maxHeap[1 << nC];
            int nS = 0;
            for (int a = 0; a < 1 << nC; a++)
            {
                sum s = {
                    .som = sumsC[a].som + sumsD[(1 << nD) - 1].som,
                    .comb = sumsC[a].comb | (sumsD[(1 << nD) - 1].comb << nC),
                    .ind1 = a,
                    .ind2 = (1 << nD) - 1};
                insertMax(maxHeap, s, nS++);
            }
            if (!schroeppel(desired, minHeap, maxHeap, nA, nB, nC, nD, sumsA, sumsB, sumsC, sumsD, nP, nS))
            {
                printf("\nSoluçao NÃO encontrada!\n");
                break;
            }

            time = cpu_time() - time;
            somaTempos += time;
            if (time > tempoMaximo)
                tempoMaximo = time;
        }
        free(sumsA);
        free(sumsB);
        free(sumsC);
        free(sumsD);

        double tempoMedio = somaTempos / n_sums;
        printf("%d %f %f\n\n", i + 10, tempoMedio, tempoMaximo);
    }
}
return 0;
}
```

Código - MATLAB

```
%RECURSIVA
valores = load("daniel2REC.txt");
n1 = valores(1:end,1);        %primeira col
tmed1 = valores(1:end,2);     %segunda col
tmax1 = valores(1:end,3);     %terceira col


%RECURSIVA - otimizada
valores = load("daniel2OTIM.txt");
n = valores(1:end,1);         %primeira col
tmed = valores(1:end,2);      %segunda col
tmax = valores(1:end,3);      %terceira col
```

```
%% COMPARAR BRUTE FORCES
% RECURSIVAS
figure(5);
semilogy(n1,tmed1,'.-');
ylabel("Tempo(s)");
xlabel("Variação de n")
title('Comparação das Brute Forces')
hold on;
semilogy(n1,tmax1, '*-')
hold on;
semilogy(n,tmed,'+-');
hold on;
semilogy(n,tmax,'o-');
legend("tempoMédio", "tempoMáx", "tempoMédioOTIM", "tempoMáxOTIM");
xlim([10 36])
grid on;
hold off;
```

```
%% BF OTIM -> 3 PCS

%DANIEL
valores = load("danielOTIM.txt");
n = valores(1:end,1);       %primeira col
tmed = valores(1:end,2);    %segunda col
tmax = valores(1:end,3);    %terceira col

valores = load("daniel2OTIM.txt");
n1 = valores(1:end,1);       %primeira col
tmed1 = valores(1:end,2);    %segunda col
tmax1 = valores(1:end,3);    %terceira col

valores = load("gui_OTIM.txt");
n2 = valores(1:end,1);       %primeira col
tmed2 = valores(1:end,2);    %segunda col
tmax2 = valores(1:end,3);    %terceira col

%
semilogy(n,tmed,'.-');
ylabel("Tempo(s)");
xlabel("Variação de n")
title('Brute Force Otimizada (Comparação 3 PCs)')
hold on;
semilogy(n,tmax, '*-')
hold on;
semilogy(n1,tmed1,'.-');
hold on;
semilogy(n1,tmax1, '*-')
hold on;
semilogy(n2,tmed2,'.-');
hold on;
semilogy(n2,tmax2, '*-')
hold on;
legend("tempoMédioPC1", "tempoMáximoPC1", "tempoMédioPC2", "tempoMáximoPC2", "tempoMédioPC3", "tempoMáximoPC3");
xlim([10 36])
grid on;
hold off;
```

```
%% BF REC -> 3 PCS

%DANIEL
valores = load("danielREC.txt");
n = valores(1:end,1);       %primeira col
tmed = valores(1:end,2);    %segunda col
tmax = valores(1:end,3);    %terceira col

valores = load("daniel2REC.txt");
n1 = valores(1:end,1);       %primeira col
tmed1 = valores(1:end,2);    %segunda col
tmax1 = valores(1:end,3);    %terceira col

valores = load("gui_REC.txt");
n2 = valores(1:end,1);       %primeira col
tmed2 = valores(1:end,2);    %segunda col
tmax2 = valores(1:end,3);    %terceira col
```

```
%
semilogy(n,tmed,'.-');
ylabel("Tempo(s)");
xlabel("Variação de n")
title('Brute Force (Comparação 3 PCs)')
hold on;
semilogy(n,tmax, '*-')
hold on;
semilogy(n1,tmed1,'.-');
hold on;
semilogy(n1,tmax1, '*-')
hold on;
semilogy(n2,tmed2,'.-');
hold on;
semilogy(n2,tmax2, '*-')
hold on;
legend("tempoMédioPC1", "tempoMáximoPC1", ...
    "tempoMédioPC2", "tempoMáximoPC2", ...
    "tempoMédioPC3", "tempoMáximoPC3");
xlim([10 36])
grid on;
hold off;
```

```
%% COMPARAR AS RECURSIVAS - 3 pcs
clear;
clc;
close all;

%PC 1
valores = load("danielOTIM.txt");
nOTIM1 = valores(1:end,1);       %primeira col
tmedOTIM1 = valores(1:end,2);    %segunda col
tmaxOTIM1 = valores(1:end,3);    %terceira col

%PC 2
valores = load("daniel2OTIM.txt");
nOTIM2 = valores(1:end,1);       %primeira col
tmedOTIM2 = valores(1:end,2);    %segunda col
tmaxOTIM2 = valores(1:end,3);    %terceira col

valores = load("gui_OTIM.txt");
nOTIM3 = valores(1:end,1);       %primeira col
tmedOTIM3 = valores(1:end,2);    %segunda col
tmaxOTIM3 = valores(1:end,3);    %terceira col
```

```
%pc1
valores = load("danielREC.txt");
n = valores(1:end,1);       %primeira col
tmed = valores(1:end,2);    %segunda col
tmax = valores(1:end,3);    %terceira col

%pc2
valores = load("daniel2REC.txt");
n1 = valores(1:end,1);       %primeira col
tmed1 = valores(1:end,2);    %segunda col
tmax1 = valores(1:end,3);    %terceira col

%pc3
valores = load("gui_REC.txt");
n2 = valores(1:end,1);       %primeira col
tmed2 = valores(1:end,2);    %segunda col
tmax2 = valores(1:end,3);    %terceira col
```

```
semilogy(n,tmed,'.-');
ylabel("Tempo(s)");
xlabel("Variação de n")
title('Brute Forces (Comparação 3 PCs)')
hold on;
semilogy(n,tmax, '*-')
hold on;
%pc2
semilogy(n1,tmed1,'.-');
hold on;
semilogy(n1,tmax1, '*-')
hold on;
%pc3
semilogy(n2,tmed2,'.-');
hold on;
semilogy(n2,tmax2, '*-')
hold on;
%pc1 OTIM
semilogy(nOTIM1,tmedOTIM1,'.-');
hold on;
semilogy(nOTIM1,tmaxOTIM1, '*-')
hold on;
%pc2 OTIM
semilogy(nOTIM2,tmedOTIM2,'.-');
hold on;
semilogy(nOTIM2,tmaxOTIM2, '*-')
hold on;
%pc3 OTIM
semilogy(nOTIM3,tmedOTIM3,'.-');
```

```
hold on;
semilogy(nOTIM3,tmaxOTIM3, '*-')
hold on;
legend("tempoMédioPC1", "tempoMáximoPC1", ...
    "tempoMédioPC2", "tempoMáximoPC2", ...
    "tempoMédioPC3", "tempoMáximoPC3", ...
    "tempoMédioPC1OTIM", "tempoMáximoPC1OTIM", ...
    "tempoMédioPC2OTIM", "tempoMáximoPC2OTIM", ...
    "tempoMédioPC3OTIM", "tempoMáximoPC3OTIM");
xlim([10 36])
grid on;
hold off;
```

```
%% 3º e 4º algor

%horowitz
valores = load("danielHORO.txt");
n1 = valores(1:end,1);       %primeira col
tmed1 = valores(1:end,2);    %segunda col
tmax1 = valores(1:end,3);    %terceira col

%schroalgçaslçglçasçg
valores = load("danielSCH.txt");
n2 = valores(1:end,1);       %primeira col
tmed2 = valores(1:end,2);    %segunda col
tmax2 = valores(1:end,3);    %terceira col
```

```
% 3 e 4
figure(5);
semilogy(n1,tmed1,'.-');
ylabel("Tempo(s)");
xlabel("Variação de n")
title('Comparação ‘Horowitz and Sahni‘ e ‘Schroeppel and Shamir‘ (PC1)')
hold on;
semilogy(n1,tmax1, '*-')
hold on;
semilogy(n2,tmed2,'+-');
hold on;
semilogy(n2,tmax2,'o-');
legend("tempoMédioHORO", "tempoMáxHORO", "tempoMédioSCHR", "tempoMáxSCHR");
xlim([10 65])
grid on;
hold off;
```

```
%% COMPARAR  3  E  4   -> 3 PCS
clear;
clc;
close all;

%horowitz
valores = load("danielHORO.txt");
nHORO1 = valores(1:end,1);       %primeira col
tmedHORO1 = valores(1:end,2);    %segunda col
tmaxHORO1 = valores(1:end,3);    %terceira col

valores = load("daniel2HORO.txt");
nHORO2 = valores(1:end,1);       %primeira col
tmedHORO2 = valores(1:end,2);    %segunda col
tmaxHORO2 = valores(1:end,3);    %terceira col

valores = load("guiHORO.txt");
nHORO3 = valores(1:end,1);       %primeira col
tmedHORO3 = valores(1:end,2);    %segunda col
tmaxHORO3 = valores(1:end,3);    %terceira col


%schroppel
valores = load("danielSCH.txt");
nSCH1 = valores(1:end,1);       %primeira col
tmedSCH1 = valores(1:end,2);    %segunda col
tmaxSCH1 = valores(1:end,3);    %terceira col

valores = load("daniel2SCH.txt");
nSCH2 = valores(1:end,1);       %primeira col
tmedSCH2 = valores(1:end,2);    %segunda col
tmaxSCH2 = valores(1:end,3);    %terceira col

valores = load("guiSCH.txt");
nSCH3 = valores(1:end,1);       %primeira col
tmedSCH3 = valores(1:end,2);    %segunda col
tmaxSCH3 = valores(1:end,3);    %terceira col
```

```
% horo 1
semilogy(nHORO1,tmedHORO1,'.-');
ylabel("Tempo(s)");
xlabel("Variação de n")
title("'Horowitz and Sahni' e 'Schroeppel and Shamir' (Comparação 3 PCs)")
hold on;
semilogy(nHORO1,tmaxHORO1,'*-');
hold on;

% horo 2
semilogy(nHORO2,tmedHORO2,'.-');
hold on;
semilogy(nHORO2,tmaxHORO2,'*-');
hold on;

%horo 3
semilogy(nHORO3,tmedHORO3,'.-');
hold on;
semilogy(nHORO3,tmaxHORO3,'*-');
hold on;

%SCH 1
semilogy(nSCH1,tmedSCH1,'.-');
hold on;
semilogy(nSCH1,tmaxSCH1,'*-');
hold on;
```

```
    %SCH 2
    semilogy(nSCH2,tmedSCH2,'.-');
    hold on;
    semilogy(nSCH2,tmaxSCH2,'*-');
    hold on;

    %SCH 3
    semilogy(nSCH3,tmedSCH3,'.-');
    hold on;
    semilogy(nSCH3,tmaxSCH3,'*-');
    hold on;

    legend("tempoMédioPC1-HORO", "tempoMáximoPC1-HORO", ...
        "tempoMédioPC2-HORO", "tempoMáximoPC2-HORO", ...
        "tempoMédioPC3-HORO", "tempoMáximoPC3-HORO", ...
        "tempoMédioPC1-SCH", "tempoMáximoPC1-SCH", ...
        "tempoMédioPC2-SCH", "tempoMáximoPC2-SCH,", ...
        "tempoMédioPC3-SCH", "tempoMáximoPC3-SCH");

    xlim([10 65])
    grid on;
    hold off;
```

```
%%
clear;
clc
close all;

%PC 1
valores = load("danielOTIM.txt");
nOTIM1 = valores(1:end,1);        %primeira col
tmedOTIM1 = valores(1:end,2);     %segunda col
tmaxOTIM1 = valores(1:end,3);     %terceira col

%PC 2
valores = load("daniel2OTIM.txt");
nOTIM2 = valores(1:end,1);        %primeira col
tmedOTIM2 = valores(1:end,2);     %segunda col
tmaxOTIM2 = valores(1:end,3);     %terceira col
```

```
valores = load("gui_OTIM.txt");
nOTIM3 = valores(1:end,1);        %primeira col
tmedOTIM3 = valores(1:end,2);     %segunda col
tmaxOTIM3 = valores(1:end,3);     %terceira col

%normais
%pc1
valores = load("danielREC.txt");
n = valores(1:end,1);        %primeira col
tmed = valores(1:end,2);     %segunda col
tmax = valores(1:end,3);     %terceira col

%pc2
valores = load("daniel2REC.txt");
n1 = valores(1:end,1);        %primeira col
tmed1 = valores(1:end,2);     %segunda col
tmax1 = valores(1:end,3);     %terceira col

%pc3
valores = load("gui_REC.txt");
n2 = valores(1:end,1);        %primeira col
tmed2 = valores(1:end,2);     %segunda col
tmax2 = valores(1:end,3);     %terceira col
```

```
%horowitz
valores = load("danielHORO.txt");
nHORO1 = valores(1:end,1);        %primeira col
tmedHORO1 = valores(1:end,2);     %segunda col
tmaxHORO1 = valores(1:end,3);     %terceira col

valores = load("daniel2HORO.txt");
nHORO2 = valores(1:end,1);        %primeira col
tmedHORO2 = valores(1:end,2);     %segunda col
tmaxHORO2 = valores(1:end,3);     %terceira col

valores = load("guiHORO.txt");
nHORO3 = valores(1:end,1);        %primeira col
tmedHORO3 = valores(1:end,2);     %segunda col
tmaxHORO3 = valores(1:end,3);     %terceira col
```

```
%schroppel
valores = load("danielSCH.txt");
nSCH1 = valores(1:end,1);         %primeira col
tmedSCH1 = valores(1:end,2);      %segunda col
tmaxSCH1 = valores(1:end,3);      %terceira col

valores = load("daniel2SCH.txt");
nSCH2 = valores(1:end,1);         %primeira col
tmedSCH2 = valores(1:end,2);      %segunda col
tmaxSCH2 = valores(1:end,3);      %terceira col

valores = load("guiSCH.txt");
nSCH3 = valores(1:end,1);         %primeira col
tmedSCH3 = valores(1:end,2);      %segunda col
tmaxSCH3 = valores(1:end,3);      %terceira col
```

```
%
semilogy(n,tmed,'.-');
ylabel("Tempo(s)");
xlabel("Variação de n")
title('Comparação Total - 3 PCs')
hold on;
semilogy(n,tmax, '*-')
hold on;
%pc2
semilogy(n1,tmed1,'.-');
hold on;
semilogy(n1,tmax1, '*-')
hold on;
%pc3
semilogy(n2,tmed2,'.-');
hold on;
semilogy(n2,tmax2, '*-')
hold on;
%pc1 OTIM
semilogy(nOTIM1,tmedOTIM1,'.-');
hold on;
semilogy(nOTIM1,tmaxOTIM1, '*-')
hold on;
%pc2 OTIM
semilogy(nOTIM2,tmedOTIM2,'.-');
hold on;
semilogy(nOTIM2,tmaxOTIM2, '*-')
hold on;
```

```
%pc3 OTIM
semilogy(nOTIM3,tmedOTIM3,'.-');
hold on;
semilogy(nOTIM3,tmaxOTIM3, '*-')
hold on;

% horo e 4
%HORO1
semilogy(nHORO1,tmedHORO1,'.-');
hold on;
semilogy(nHORO1,tmaxHORO1,'*-');
hold on;

% horo 2
semilogy(nHORO2,tmedHORO2,'.-');
hold on;
semilogy(nHORO2,tmaxHORO2,'*-');
hold on;

%horo 3
semilogy(nHORO3,tmedHORO3,'.-');
hold on;
semilogy(nHORO3,tmaxHORO3,'*-');
hold on;
```

```
%SCH 1
semilogy(nSCH1,tmedSCH1,'.-');
hold on;
semilogy(nSCH1,tmaxSCH1,'*-');
hold on;

%SCH 2
semilogy(nSCH2,tmedSCH2,'.-');
hold on;
semilogy(nSCH2,tmaxSCH2,'*-');
hold on;

%SCH 3
semilogy(nSCH3,tmedSCH3,'.-');
hold on;
semilogy(nSCH3,tmaxSCH3,'*-');
hold on;
```

```
legend("tempoMédioPC1", "tempoMáximoPC1", ...
    "tempoMédioPC2", "tempoMáximoPC2", ...
    "tempoMédioPC3", "tempoMáximoPC3", ...
    "tempoMédioPC1OTIM", "tempoMáximoPC1OTIM", ...
    "tempoMédioPC2OTIM", "tempoMáximoPC2OTIM", ...
    "tempoMédioPC3OTIM", "tempoMáximoPC3OTIM", ...
    "tempoMédioPC1-HORO", "tempoMáximoPC1-HORO", ...
    "tempoMédioPC2-HORO", "tempoMáximoPC2-HORO", ...
    "tempoMédioPC3-HORO", "tempoMáximoPC3-HORO", ...
    "tempoMédioPC1-SCH", "tempoMáximoPC1-SCH", ...
    "tempoMédioPC2-SCH", "tempoMáximoPC2-SCH,", ...
    "tempoMédioPC3-SCH", "tempoMáximoPC3-SCH");
xlim([10 65])
grid on;
hold off;
```